

University of Southern Queensland
Faculty of Engineering and Surveying

**Supervisory Control of Material Handling Conveyor
Belt System**

A dissertation submitted by
Mr. Reec Rodolfo Muorwel

In fulfilment of the requirements of
Courses ENG4111 and 4112 Research Project

Towards the degree of
Bachelor of Engineering (Electrical and electronic)

Submitted: October, 2008

Abstract

In these modern days, there are large increases in technology in which the controlling and monitoring of the system can be achieved using software such as PLC and SCADA. The functions and applications are well discussed in this dissertation.

This project has involved the standard industrialised and process engineering softwares which are able to solve very complex and challenging tasks. Both PLC and SCADA are mostly used in production, manufacturing, power generation, refining, and fabrication and process industries.

This dissertation will deeply outline the role of PLC in the Conveyor belt system to achieve the sorting of the different types of washer (material type). It will also give the steps that are needed to achieve the controlling and monitoring of the conveyor belt system using SCADA. Their theoretical backgrounds will be discussed as well as the role played by the optical sensors and actuators in this project.

University of Southern Queensland

Faculty of Engineering and Surveying

**ENG4111 Research Project Part 1 &
ENG4112 Research Project Part 2**

Limitations of Use

The Council of the University of Southern Queensland, its Faculty of Engineering and Surveying, and the staff of the University of Southern Queensland, do not accept any responsibility for the truth, accuracy or completeness of material contained within or associated with this dissertation.

Persons using all or any part of this material do so at their own risk, and not at the risk of the Council of the University of Southern Queensland, its Faculty of Engineering and Surveying or the staff of the University of Southern Queensland.

This dissertation reports an educational exercise and has no purpose or validity beyond this exercise. The sole purpose of the course pair entitled "Research Project" is to contribute to the overall education within the student's chosen degree program. This document, the associated hardware, software, drawings, and other material set out in the associated appendices should not be used for any other purpose: if they are so used, it is entirely at the risk of the user.



Professor Frank Bullen
Dean
Faculty of Engineering and Surveying

Certification

I certify that the ideas, designs and experimental work, results, analysis and conclusions set out in this dissertation are entirely my own efforts, except where otherwise indicated and acknowledged.

I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.

Reec Rodolfo Muorwel

Student Number: 0050043061

Signature

30/10/2008 Date

Acknowledgements

I would like extend my gratitude to my supervisor Dr. Tony Ahfock for his assistance during this project.

I would also like to thank Don Gelhaar of USQ technical staff for providing me access to the laboratory throughout the year.

ABSTRACT	I
LIMITATIONS OF USE	II
CERTIFICATION	III
ACKNOWLEDGEMENTS	IV
LIST OF FIGURES	VII
LIST OF TABLES	VIII
CHAPTER 1	1
1.0 INTRODUCTION	1
1.1 PROJECT JUSTIFICATION	1
1.2 PROJECT OBJECTIVES	2
1.3 METHODOLOGY	3
1.4 OUTLINE OF DISSERTATION	5
CHAPTER 2	7
2.0 THE CONVEYOR SYSTEM	7
2.1 OVERVIEW	7
2.1 HARDWARE DESCRIPTION	7
2.1.1 Photoelectric Sensor	8
2.1.2 Proximity Sensors	9
2.1.3 Inductive Proximity and Induction of Eddy Current	9
2.1.4 Optical Sensor	9
2.1.5 Actuator	10
2.2 SEQUENCE OF OPERATION	12
2.2.1 Dispenser	12
2.2.2 Initial Sensor	13
2.2.3 Height Measurement System	13
2.2.4 Second Optical and Collection Bins	14
CHAPTER 3	16
3.0 BASICS OF PLC PROGRAMMING	16
3.1 PLC HARDWARE AND SOFTWARE	17
3.2 AVAILABLE FEATURES IN MODERN PLCs	19
3.2.1 Counters	20
3.2.2 Timers	21
3.3 LADDER DIAGRAM	23
3.4 PROGRAMMING SOFTWARE AND PROGRAMMING ENVIRONMENT	23
CHAPTER 4	34
4.0 SUPERVISORY CONTROL AND DATA ACQUISITION	34

4.1 INTRODUCTION	34
4.1.1 SCADA hardware	35
4.1.2 Communication between PLC and SCADA	35
4.2 FEATURES OF CITECT SCADA	38
4.2.1 User interface include	38
4.2.2 Alarms	39
4.2.3 Trends	40
4.2.4 Scalability	41
4.2.5 Access to data	41
4.2.6 Database	42
4.2.7 Networking	42
4.2.8 Fault tolerance and redundancy	43
4.2.9 Client/ server distributed processing	43
CHAPTER 5	45
5.0 CONVEYOR BELT CONTROL USING THE SIEMENS S7200 PLC BASED CONTROLLER	45
5.1 INTRODUCTION	45
5.2 INPUT AND OUTPUT REQUIREMENTS	46
5.3 LADDER DIAGRAM	49
5.3.1 Main Program	49
5.3.2 Subroutine 0 (SBR_0)	66
5.3.3 Subroutine 1 (SBR_1)	69
5.3.4 Subroutine 2 (SBR_2)	74
5.4.1 INPUTS	78
5.4.2 OUTPUTS	78
5.4.3 Subroutines	79
5.4.5 First optical sensor	80
5.4.6 Second Sensor	81
5.4.7 Manual Demonstration Procedures for Conveyor Belt Subsystem	81
CHAPTER 6	83
6.0 SCADA SYSTEM FOR THE CONVEYOR	83
6.1 INTRODUCTION	83
6.2 THE SETUP OF COMMUNICATION BETWEEN PLC AND SCADA	83
6.2.1 Express Communications Wizard	83
6.2.2 Communications Dialogs	90
6.2.3 Testing Communications	90
CHAPTER 7	98
7.0 DISCUSSION AND CONCLUSION	98
7.1 ACHIEVEMENTS OF OBJECTIVES	98
7.2 FURTHER WORK	99
7.3 CONCLUSION	100
REFERENCES	101
APPENDIX A- PROJECT SPECIFICATIONS	103
APPENDIX B- PLC LADDER DIAGRAM	104
APPENDIX C- DEMONSTRATION SCENARIOS	119

List of figures

FIGURE 2.1 (A) GENERAL VIEW OF ELECTROMAGNETIC ACTUATOR	11
FIGURE 2.1 (B) A LAYOUT OF ELECTROMAGNETIC ACTUATOR	11
FIGURE 2.2 (A): DISPENSER.....	12
FIGURE 2.2(B): OPTICAL SENSOR.....	13
FIGURE 2.2(C): HEIGHT MEASUREMENT GAUGE.....	14
FIGURE 2.2 (D): COLLECTION BINS	15
FIGURE 3.0 PROGRAMMABLE LOGIC CONTROLLERS	17
FIGURE 3.1 (A)	18
FIGURE 3.1 (B)	19
FIGURE 3.4(A).....	26
FIGURE 3.4 (B)	27
FIGURE 3.4 (C).....	28
FIGURE 3.4 (D).....	29
FIGURE 3.4 (E)	30
FIGURE 3.4 (F).....	31
FIGURE 4.1 (A).....	36
FIGURE 4.1 (B)	37
FIGURE 4.1 (C).....	37
FIGURE 6.2 (A) EXPRESS COMMUNICATION WIZARD	85
FIGURE 6.2 (B) EXPRESS COMMUNICATION WIZARD WITH I/O SERVER NAME	85
FIGURE 6.2 (C) EXPRESS COMMUNICATION WIZARD WITH EDIT I/O DEVICE OPTION	86
FIGURE 6.2 (D) EXPRESS COMMUNICATION WIZARD WITH DIFFERENT I/O DEVICE	87
FIGURE 6.2 (E) MANUFACTURE, MODEL AND METHOD OF COMMUNICATION	87
FIGURE 6.2 (F) PUBLIC SWITCHED TELEPHONE NETWORK OPTION	88
FIGURE 6.2 (G) SERIAL PORTS DETECTED ON SYSTEM.....	89
FIGURE 6.2 (H) EXTERNAL TAG DATABASE OPTION	89
FIGURE 6.2 (I) FINAL STEP FOR EXPRESS COMMUNICATION WIZARD.....	90
FIGURE 6.2 (J) VARIABLE TAG FOR COMMUNICATION TEST.....	91
FIGURE 6.2 (K) BUTTON PROPERTIES FOR COMMUNICATION TEST.....	92
FIGURE 6.2 (L) BUTTON PROPERTIES UP COMMAND FOR COMMUNICATION TEST	93
FIGURE 6.2 (M) TEXT PROPERTIES FOR COMMUNICATION TEST.....	94
FIGURE 6.2 (N) MIMIC DIAGRAM FOR COMMUNICATION TEST	95

List of Tables

TABLE 2.1 ATTRIBUTES 10

TABLE 3.2: SIEMENS S7200 SPECIFICATION FOR THE TIMERS 22

TABLE 3.4(A): BAUD RATE SUPPORTED BY S7-200 24

TABLE 3.4(B): DEFAULT ADDRESSES FOR S7-200 DEVICES..... 25

TABLE 3.4(C); NUMBER OF CONNECTION FOR THE S7-200 CPU AND EM277 MODULES..... 32

Chapter 1

1.0 Introduction

1.1 Project Justification

Modernity has advanced in the most areas of engineering particularly, in the areas of manufacturing engineering, control engineering, process/chemical engineering, environmental engineering and many other disciplines in engineering. All the above mentioned disciplines have roles to play in the society.

Programmable logic control also known as PLC is mostly used in the mentioned areas above to achieve and solve complex and challenging tasks in engineering and technology, generally in the automatic systems such as the control of assembly line machinery in factories. Important software that is essentially used in industries is the Supervisory Control and Data Acquisition (SCADA). SCADA is mostly used in the areas that deal with autonomous Remote Terminal Unit (RTU). It is used for production, manufacturing, power engineering, refining, fabrication, process and many other areas.

The aim of this project is to sort different washers and determine whether they are plastic or metal. The unwanted washer that did not meet relevant criteria is disposed into a rejection bin. The sorting of these washers include the uses of the electromagnetic actuators as well as the optical sensors. The functions of electromagnetic actuators mainly include the dispensation of the washer and also controlling the bins that the washers are directed to. The sensors however, are used to monitor the incoming washer and detect the washers' material type.

1.2 Project Objectives

The specified objectives for this project are:

- To become familiar and confident with basic concepts of ladder logic
- To convert the existing Allen Bradley Micrologix 1000 PLC based controller into Siemens S7200 PLC based Controller.
- To familiarise with the Citect supervisory control and data acquisition software(SCDA)
- To construct a mimic diagram using Citect to allow operation of the conveyor system from a PC.

1.3 Methodology

This project has being done previously using Allen Bradley PLC based. The procedures involved in this project prior to the completion are summarised below:

- Replaced the existing Allen Bradley Micrologix 1000 PLC based controller with new standard PLC (Siemens S7200 PLC based controller)

The operation of the previous PLC needs careful understanding as well as the connection of the previous PLC. This wills more knowledge and more understanding of how PLC works.

- Understand the general operation of ladder logic in Siemens S7200 PLC based controller
The understanding should include all the required knowledge about PLC. This should include the understanding of timers and their different usage such as on/off time and their specifications as well. Understanding of counters such as count up, count down, count up/down and how they operate, including all the inputs, resets and power supplies are necessary. There is also need of building the confident and understanding of how the inputs (I), outputs (Q), logic operators, special memory etc.

- Convert the existing Allen Bradley Micrologix 1000 ladder logic programme to Siemens s7200 ladder logic programme.

This process is one of the main objectives in this project therefore, their understanding is valuable. Both have timers, counters, inputs and outputs but they are different in configuration.

- Test the written program

The written program is tested and documented prior toward the supervision of SCADA.

- Understand Citect SCADA and mimic diagram

After the completion of PLC testing, SCADA should be incorporated therefore; the general operation of SCADA should be understood. This includes the understanding of how SCADA work and how to build a mimic diagram.

- Build Citect SCADA mimic diagram and test

Successful mimic diagram should be able to control the system from the personal computer using keyboards, mouse etc.

- Write up dissertation

This is the last step required in which all information has to be gathered, grouped and presented or reported in a very professional manner(dissertation). Dissertation should include background, diagrams, tables, figures and other important information that are needed in this project.

1.4 Outline of Dissertation

This dissertation is comprised of seven chapters and which the first six chapters focused mainly on the objectives set out in the first chapter (Introduction Chapter). The overview of each chapter will be briefly summarized below for simplicity to the reader.

- Chapter 1 Gives the introduction and presentation of the objectives and methodology of the project.
- Chapter 2 This chapter provides the overview, sequence of operation and hardware description of the conveyor system. The theoretical background of optical sensors and actuators are also outlined in this chapter.
- Chapter 3 This chapter discusses the basics of PLC programming. This includes the theory and background of PLC hardware and software. The available features in modern PLCs are also discussed in details. The description of Ladder logic is also provided as well as the programming software and programming environment which are described in detail.
- Chapter 4 Focuses on the Supervisory Control and Data acquisition (SCADA). It provides the general description of SCADA and features Citect that are used to control the system. For simplicity to the reader, simple examples are provided to illustrate the function of SCADA.
- Chapter 5 Discusses the operation of the conveyor belt system using the Siemens S7200 PLC based controller. It also gives details for the required inputs, outputs and description of Ladder logic for the conveyor belt system. It also provides the testing procedures that need to be followed since the control of this system is demonstrated manually, using switches.

- Chapter 6 Provides the details of how SCADA is used to control the conveyor belt system. It describes the role of HMI (human machine interface) and the general operation of the system.
- Chapter 7 This final chapter concludes the overall work of the project, discusses the achievements of the objectives and the remaining work that was not achieve due to a number of obstacles.

Chapter 2

2.0 The Conveyor System

2.1 Overview

The Conveyor belt subsystem is used for sorting different washers. The system is comprised of different sequences of operations that are needed to be accomplished prior to the final determination of the type of material. The sequence of operations in this system includes dispensing a washer, detecting the incoming washer, make an appropriate height measurement, detecting the type of the material and finally, placing the washers into their respected bins. Two optical sensors are used in this project for detection of the incoming washer as well as the detection of the type of material. Electromagnetic actuators (figure 2.1(a)) are used for dispensing the washer onto the moving conveyor belt as well as energising the right bins for the washers and the mortised height measurement gauge to perform the height measurement.

2.1 Hardware Description

As mentioned is the conveyor system overview, this section will examine the theory behind the control of this system. There are also many emphases on the sensors as they play a large role in the detection of the incoming washer as well as the detection of material type. This section will discuss the theory of photoelectric sensor, proximity sensors, the induction of eddy current, optical sensors and finally the electromagnetic actuator.

2.1.1 Photoelectric Sensor

In these modern days, sensors have been playing very important roles in engineering and technology in which they are used to control and measure an automatic system (S. Middelhoek, J. Van Der Spiegel, 1987). Their usefulness contributes to many parts of engineering. (Sabrie Soloman, 1998) stated that in order to successfully automate a process, it is always obvious to collect information about the status. Thus, sensors are used as part of the control system that is responsible for obtaining and preparing process status data as well as passing it onto a processor. As for clarification, photoelectric uses light to detect the object either it is present or absent. Photoelectric sensors use three methods for detection namely; through-beam detection, reflex detection and proximity detection.

Through-beam method needs the source and detector to be positioned opposite each other in which light can then be sent directly from source to detector. Therefore, whenever an object passes between them, the beam is broken and hence, signalling the detection of an object. Nevertheless, reflex detection method requires both source and detector to be put at the same side of the object which has to be detected. Thus, it detects when the light beam is transmitted from the source to a retro-reflector which will reflect the light to the detector. This method is mostly used due to its flexibility and simple installation as well as providing the best cost-performance ratio of these three methods. As for proximity method, the source and detector are installed on the side of the object which is to be detected and focus to the point in front of the sensor. Therefore, if an object passes in front of the source and detector, the light from the source is reflected from the surface of an object back to the detector and thus, the object is detected. These methods are widely used when dealing with light detectors. For example, initial optical detector as mentioned in the project background uses the through-beam method to detect the incoming washer and alert the motor to stop the conveyor belt from running.

2.1.2 Proximity Sensors

Proximity sensing is a method used for detecting the presence or absence of an object using an electronic non contact sensor (Sabrie Soloman, 1998). Inductive proximity sensors are used for switches that use non contact sensing metallic objects. Capacitive proximity sensors can also behave as inductive but they only detect the non-metallic objects. Both of these sensors have limit switches of up to 100mm.

2.1.3 Inductive Proximity and Induction of Eddy Current

Inductive proximity sensors consist of four basic elements. These include sensor coil and ferrite coil, Oscillator circuit, detector circuit and solid-state output circuit. Firstly, the radio-frequency electromagnetic fields are generated by the oscillator coil which are radiated from the ferrite coil and coil assembly. These fields are centred around the axis of the ferrite core hence, shaping the fields and direct them at the sensor face. Thus eddy current is induced into the surface of the target when the metal targets the field (Sabrie Soloman, 1998).

2.1.4 Optical Sensor

(J. Fluitamn and TH. Popma, 1986) strongly argue that optical sensors are always consistent with light as the source as well as the optical components to lead the light in proper shape to some modulating agent and detector lastly and may possibly proceed by some optical components. This can be proven by our own eyes because we as human are being able to detect shapes and colour movement. Therefore, the initial optical sensor in this project uses this theory to achieve

the detection of the incoming washer and notifies the motorised height measurement gauge to be brought down for performance of washer height measurement.

2.1.5 Actuator

Actuators are digital outputs controllers which turn control process variables or machine action. These include motors, solid state relay, solenoids and contactors. Generally, Solenoids contain a coil that has iron circuit which forms the fixed part. (Crispin, 1990) defined the solenoid as an electromagnetic device that converts an electrical signal into mechanical movement. As it can be from figure 2.1(a), when energising the thrust pin, it pulls the moving iron plunger into the coil. This pull process is achieved to a pushing process by fitting a suitable thrust pin as in figure 2.1(b) refer to table 2.1 for the attributes of R-S 349-709 which is used in this project.

Attributes type	Attribute value
Action	Pull
Supply Voltage	12Vdc
Closed Power Continuous	10W
Duty Cycle	100%
Maximum Stroke Length	27mm
Operating temperature Range	-5 to 55 degree Celsius
Weight	213g
Body Depth	59.2mm
Continuous Magnetic Force	400g at 5mm
Width	W32mm

Table 2.1 Attributes

Source: R-S Actuators manufacturing webpage

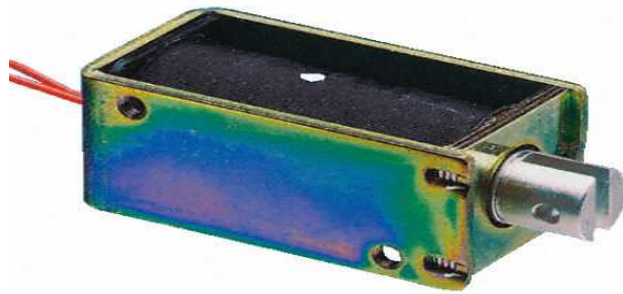


Figure 2.1 (a) general view of electromagnetic actuator
Source: R-S Actuators manufacturing webpage

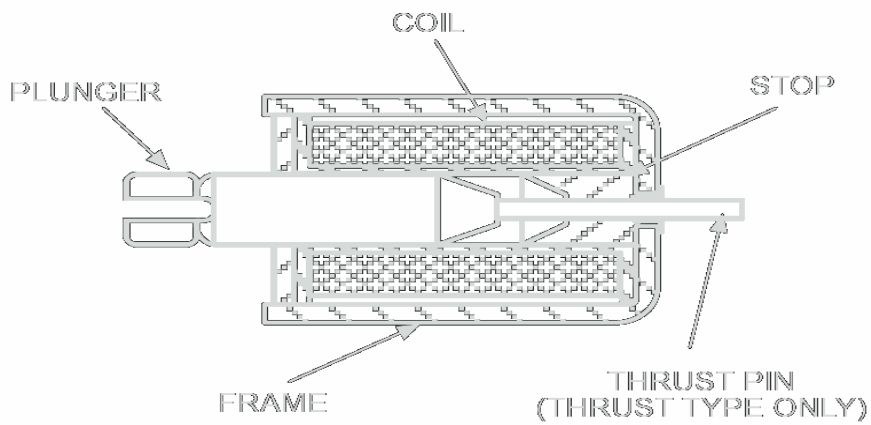


Figure 2.1 (b) a layout of electromagnetic actuator
Source: R-S Actuators manufacturing webpage

2.2 Sequence of Operation

As mentioned in the project overview, the controlling of the conveyor and sorting of the washers are achieved by sequence of events. This section outlines the importance of these sequences of operation.

2.2.1 Dispenser

An incoming washer can be dispensed by energising the electromagnetic actuator hence, pulling the armature to allow one washer to be pushed down a ramp onto a moving conveyor. At that stage, the armature will return to an un-energised position to block the next washer until it is called for the next cycle (figure 2.2(a)).

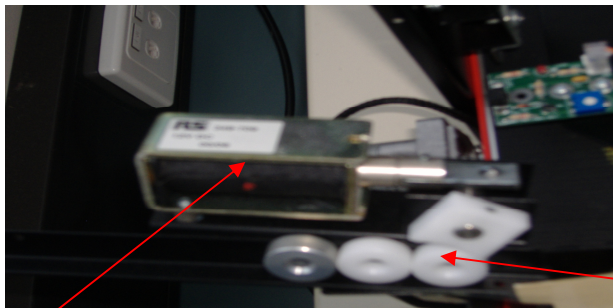


Figure 2.2 (a): dispenser

Incoming

Electromagnetic

2.2.2 Initial Sensor

The theoretical backgrounds of proximity or optical sensors are pilot devices that detect the presence of an object without physical contact (Petruzella, 1998). They are solid-state electronic devices which are capable of protecting against any chemical, excessive vibration, liquids etc. These sensors are commonly used in areas such as sensing non-metal (glass, plastic), counting, detecting the height or weight etc.

This initially uses optical sensors which will detect the incoming washer from the dispenser and alerting the motor to stop the conveyor at the right time to perform the height measurement. It is for clarification that anything that has light will be detected and that is why the system uses the black conveyor belt to avoid the interferences with the system (figure 2.2(b)).

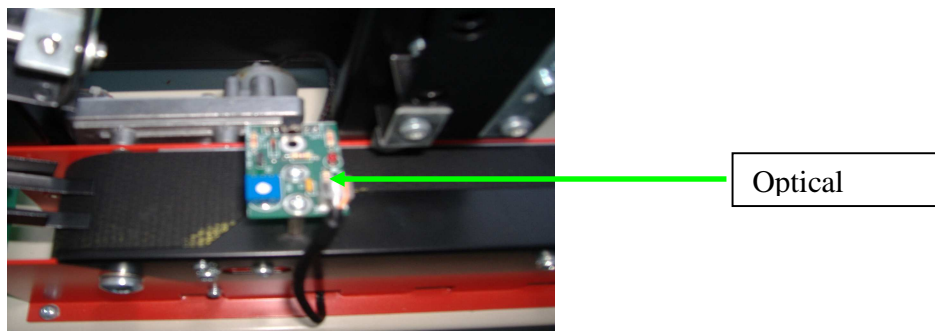


Figure 2.2(b): optical sensor

2.2.3 Height Measurement System

The performance of height measurement of the washer is the second step. This process is achieved by energising the motor to rotate hence, bringing down the motorised arm to measure the height of the washer. As for more elaboration, four LEDs are used to indicate either the motorised height measurement gauge is at the Top Dead Centre (TDC), Bottom Dead Centre

(BDC) or if the washer is either too small or too big. In other words, the height is not within the specified ranges (figure 2.2(c)).

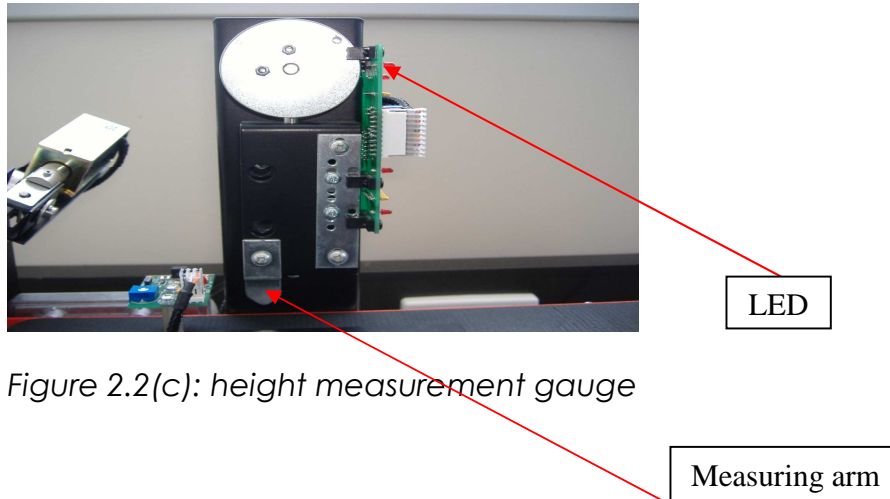


Figure 2.2(c): height measurement gauge

2.2.4 Second Optical and Collection Bins

Depending on the output of the thickness measurement system and metal detection, the washer automatically goes to one of their respected bins. The sorting of the washer if it is either metal or plastic is achieved when the metal washer passes the optical sensor thus inducing eddy current. This induced current will cause a reduction to the impedance as proved below using ohm's law.

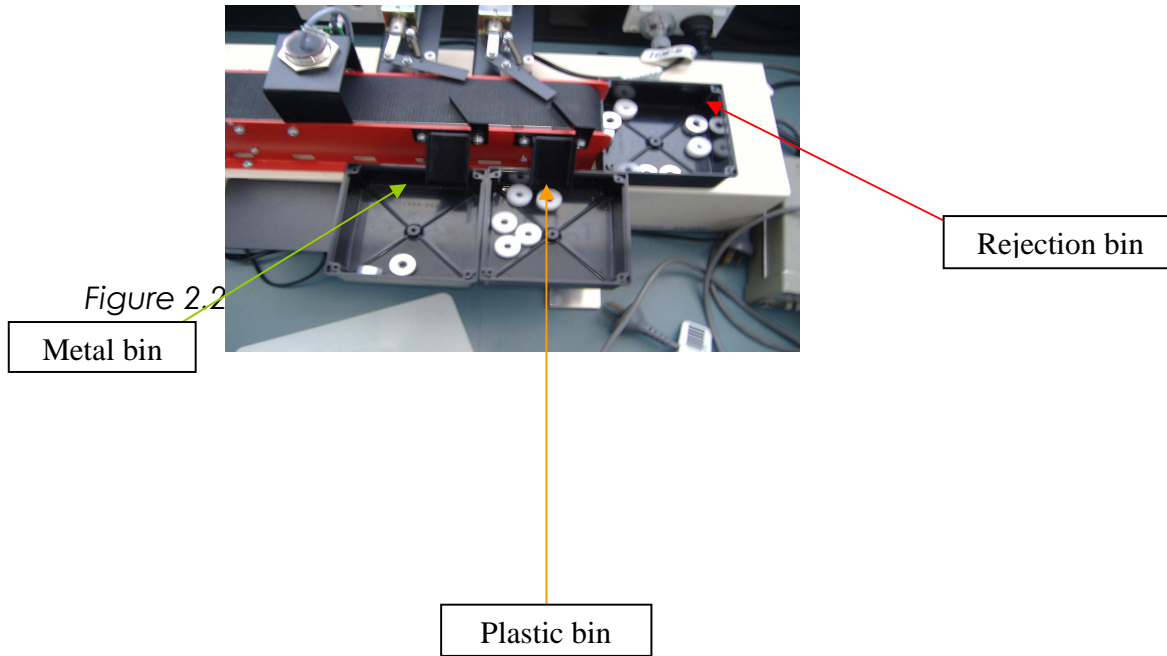
$$V = I \times Z$$

Where V is voltage in the coil

I is the current and Z is Impedance

Therefore, when the sensors see less impedance, it will recognise the washer as metal, or the washer as plastic. This will allow the electromagnetic actuator of metal to be energised while de-

energising the plastic bin and Vice-versa. The washers that are not within the tolerance will be dumped into the rejection bin. (Figure 2.2 (d))

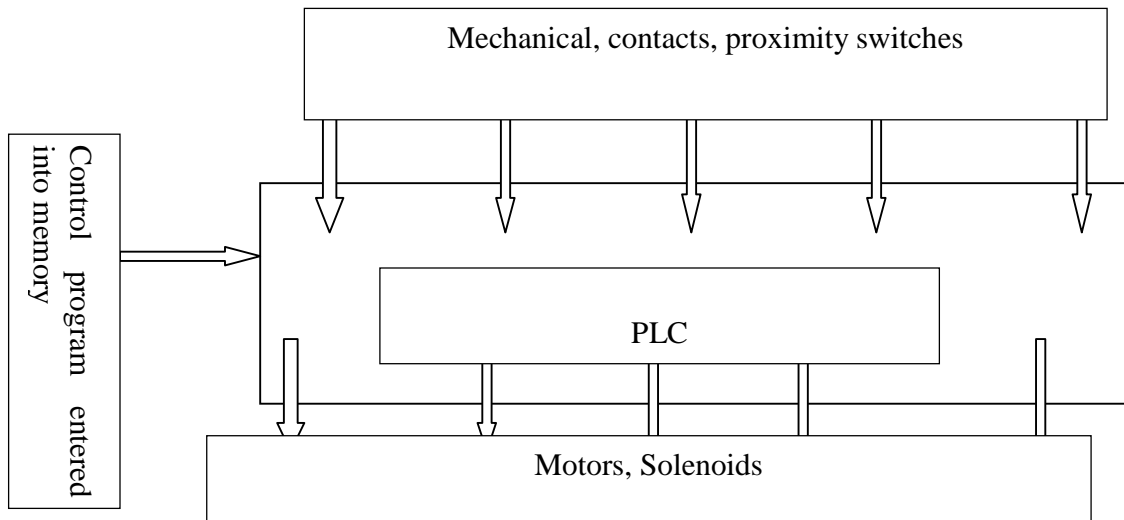


Chapter 3

3.0 Basics of PLC Programming

Programmable Logic Controller (PLC) is a digital electronic device that uses programmable memory to store instruction and to implement specific functions such as logic, sequencing, timing, counting, and arithmetic, data manipulation and communication in order to control industrialised machines and processes (Bolton, 2006). Refer to figure 3.0 for demonstration of how the control of the system is achieved. The system can be controlled by connecting both the inputs and outputs devices to the PLC. This connection will allow the user to monitor the inputs and outputs accordingly to the user defined program. Consider the term logic is frequently used as the programming is based on the logic demand of input devices (Crispin, 1990). The PLC software is used to develop the ladder diagram which can then be downloaded into the PLC through port RS232. The required inputs such as switches, sensors and push buttons are connected into the input modules however; the output modules can be controlled by the CPU of PLC. This program can executed using the RUN mode.

INPUTS



OUTPUTS

Figure 3.0 programmable logic controllers

Source: Crispin, 1990

3.1 PLC Hardware and Software

A PLC system is comprised of basic functional components such as; Processor units, memory, power supply units, the section of inputs and outputs interface, communication interface and programming device (Bolton, 2006). For more elaboration, the processor unit also known as Central Processing Unit (CPU) has a microprocessor which interrupts the inputs signal and performs the action of the control accordingly to the stored program into its memory and makes decisions for the output signal. Figure 3.1(a) below shows the new standard PLC Siemens S7200 micro PLC. The developed program is downloaded into this PLC to allow the control of the system using the required input devices.

The S7200 PLC CPU consists of a lot features that help the users to perform any kind of application. These features will be discussed later in section 3.2. Figure 3.1(b) shows the connection of S7200 CPU 224XP used in this project. It is comprised of 14 inputs and 10 digitally controlled outputs with data and program memory of 8192 bytes and an addition of 5120 bytes (petal, 2008).

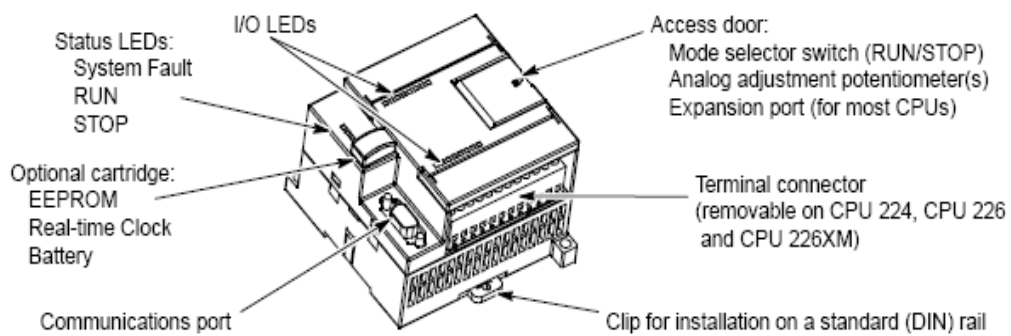


Figure 3.1 (a)

Source: (Petal, 2008)

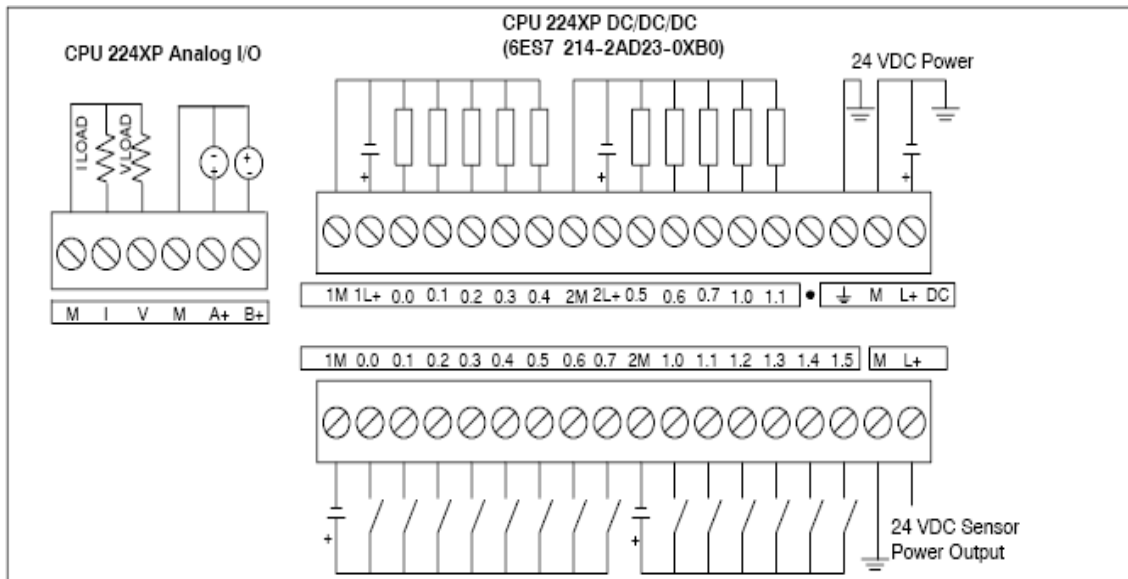


Figure 3.1 (b)

Source (Petal, 2008)

3.2 Available Features in Modern PLCs

The Siemens Simatic S7200 PLC based uses STEP 7 micro/win 32 and version 4.0 software. This software is only supported by Windows 95, windows 98 and NT environment but not supported by Windows XP. This modern PLC is consistent with a lot of features that enable the user to achieve the main aim of the project. The most commonly features used in this project will be discussed in detail. This includes counters, timers, logical operation, the input and output and, the subroutine and interrupt. Both of these will be described in detail as well as their applications.

3.2.1 Counters

The counters that are used in the PLC behave similarly to those of mechanical counters. The control of the functions of circuitry is achieved by counters in which they compare the accumulated values with the present values. Counters are mostly used to cause an event to occur until the count reaches a present value and also to count to a present value and cause an event to occur. Counters are represented in ladder logic in form of boxes (figure?). The Siemens S7200 PLC uses three types of counters namely; up counter (CTU), down counter (CTD) and up/down counter (CTUD). There are 256 counters available in the S7200 starting from C0 to C255.

The up counter counts from the current value to present value (PV). Each time the count input (CU) goes from logic 0 to logic 1, the counter is incremented by 1. This counter can then be reset by reset input (R). The counted value is stored in the present value (PV) and thus, whenever the count is equal or greater the present value, the digital output bit (Q) will be energised to be on.

The function of down counter is that it counts from the present value (PV) whenever the transition of CD goes from logic 0 to logic 1. The output (Q) is on when the current value is equal to zero. The counter can then be reset and loads the current value with the present value when the load input (LD) is enabled.

The up/down counter uses both concepts of up counter and down counter. It counts up from the present value whenever the transition of CU goes from logic 0 to logic 1 and likewise, count down when there is an occurrence of transition at CD. The output of down counter is on when the current value is equal to zero. The counter loads the current value with the present value when the load input (LD) is also enabled. Also the counter rests, and loads the current value with zero when the reset (R) is enabled. The counter stops counting when it reaches present or zero.

A good example for this application of counters is keeping the number of tracks in the parking lot. The counter can be used to count down the number of tracks entering the park until the required limits is reached, it will put the sign at the entry gate in which no more tracks are allowed. The up/down counter can be used to monitor the outgoing tracks and hence, gives space to the next tracks to enter.

3.2.2 Timers

The timers permit the delay for the certain control of operation. They are used for different purpose in the control of the system. Timers are comprised of different functions such as off-delay (TOF), On-delay (TON) and Retentive On-delay (TONR) comprised with 2566 timers.

The off-delay timer is used for delaying an output which is turning off for a certain fixed period of time after the input turns off. The timer bit can be turned on immediately after the enabling inputs are turned on. When the input turns off, the timer counts until the present time has elapsed, then the timer bit turns off. Refer to table for the resolutions available of the off-delay timer. The on-delay timer receives logic 1 or enable at its input (IN), a present amount of time (PT) passes before the timer bits turns on. The timer resets to the starting time when the enabling input (IN) goes to a logic zero. The function of Retentive on-delay performs the same function as on-delay timer with single different. The retentive on-delay timer times as long as the enabling inputs are on but does not reset when the input goes off. This timer must be reset with a Reset (R) instruction.

Timer type	Resolution	Maximum value	Timer Number
TONR	1ms	32.767 seconds	T0, T64
	10ms	327.67 seconds	T1-T4, T65-T68
	100ms	3276.6 seconds	T5-T31, T69-T95
TON,TOF	1ms	32.767 seconds	T32, T96
	10ms	327.67 seconds	T33-T36, T97-T100
	100ms	3276.7 seconds	T37-T63, T101-T255

Table 3.2: Siemens S7200 specification for the timers

Source: Siemens Schematic S7-200 manual

3.3 Ladder Diagram

A Ladder diagrams logical configuration is used for the PLC that is equivalent to that of an electrical wiring diagram. The ladder diagram is comprised of contacts, coils and a special function of blocks. Multi-connection of more than one coil is possible in ladder diagram.

The arrangement of a ladder diagram is such that it is series horizontal comprised with inputs also known as contactors and the output also known as coils. It resembles logical operation in which the inputs do precede the outputs and it is often in a form of normally open and normally closed (figure?). It must always contain at least one output. Other elements such as counters, timers, shift registers, interrupts, and subroutines can be also are implemented as part of the ladder diagram.

As mentioned previously, the term ladder is applied because the lines of the completed diagram look similar to the rungs of ladder. The ladder diagram is capable of being translated directly in to a form of program consisting of instructions and data that describes the Boolean instructions such as LOAD, LOAD NOT, AND NOT and many others.

As part of ladder diagram the operation of latching circuit is very important. This configuration is used to hold the energised coil (output) even though the contact that energised this output is in its OFF state.

3.4 Programming Software and Programming Environment

There is need to understand the basics of S7200 network communications including the uses of master and slaves devices on a network, setting the baud rate and network address, setting the remote address and the selection of the communications protocol for the network.

Master-slave network is supported by the S7200 which functioned as master or slave in the network meanwhile, the STEP7 –micro/Win is always a master. The function of the master

device in the network is to send a request to another device on the network as well responding to the request from other master on the network. Some of the master devices include STEP-Micro/WIN, HMI devices such as TD 200, S7-300 or S7-400 PLCs. Nevertheless, the slave devices can only respond to the request made from the master devices. S7200 functions in as a slave in most networks.

Apart from the setup of master-slave and slave devices for the network, the baud rate and network address also needs to be set. Baud rate is the speed for transmitting the data across the network and it is typically measured in kilobaud or megabaud. It is the measured amount of data that can be transmitted within the fixed amount of time. Considerably, each device that communicates over a particular network must be configured to transmit data at the same baud rate. And thus, the slowest device that is connected to the network determines the fastest baud rate for the network. The baud rates that are supported by the S7200 are listed in the table 3.4(a) below. In addition, the network address is defined as the unique number that is assigned to particular device on the network. This number must ensure that there is transfer and retrieve of the data from the correct device. Network addresses from 0 to 126 are supported by S7200 as well for an S7200 with two ports; there is network address for each port. Refer to the table 3.4(b) for the detail of the default setting of the S7200 devices.

Network	Baud Rate
Standard network	9.6kBaud to 187.5kBaud
Using an EM 277	9.6kBaud to 12Mbaud
Freeport	1200Baud to 115.2kBaud

Table 3.4(a): Baud rate supported by S7-200

Source: CITECT SCADA Help

S7-200 device	Default address
STEP-7 Micro/WIN	0

HMI(TD 200,TP or OP)	1
S7-200 CPU	2

Table 3.4(b): Default addresses for S7-200 devices

Source: CITECT SCADA Help

As discussed above, this section gives the detail of setting up the baud rate as well as the network address for the STEP-7 Micro/WIN. Configuration of the baud rate and network address for the STEP-7 Micro/WIN is critical. The baud rate must be the same as the other devices on the network while the network address must be a unique number. The configuration of the baud rate and network address is achieved using the below steps which are also displayed in the figure 3.4 (a) below. These figures are found from Siemens Schematic S7-200 manual

1. Double click the icon in the communication setup window
2. Click the properties button on the set PG/PC interface dialog box
3. Select the network address for STEP-7 –Micro/WIN
4. Select the baud rate for STEP-7—Micro/WIN

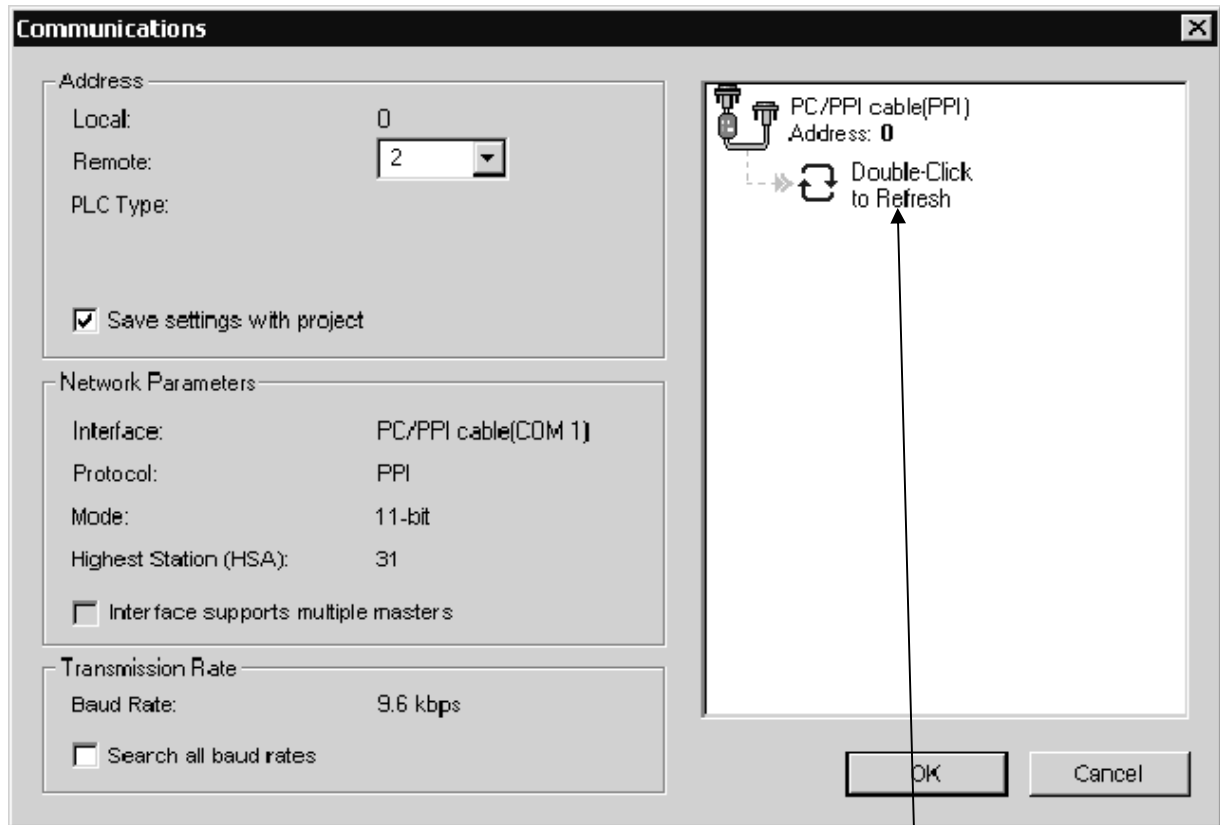


Figure 3.4(a)

1

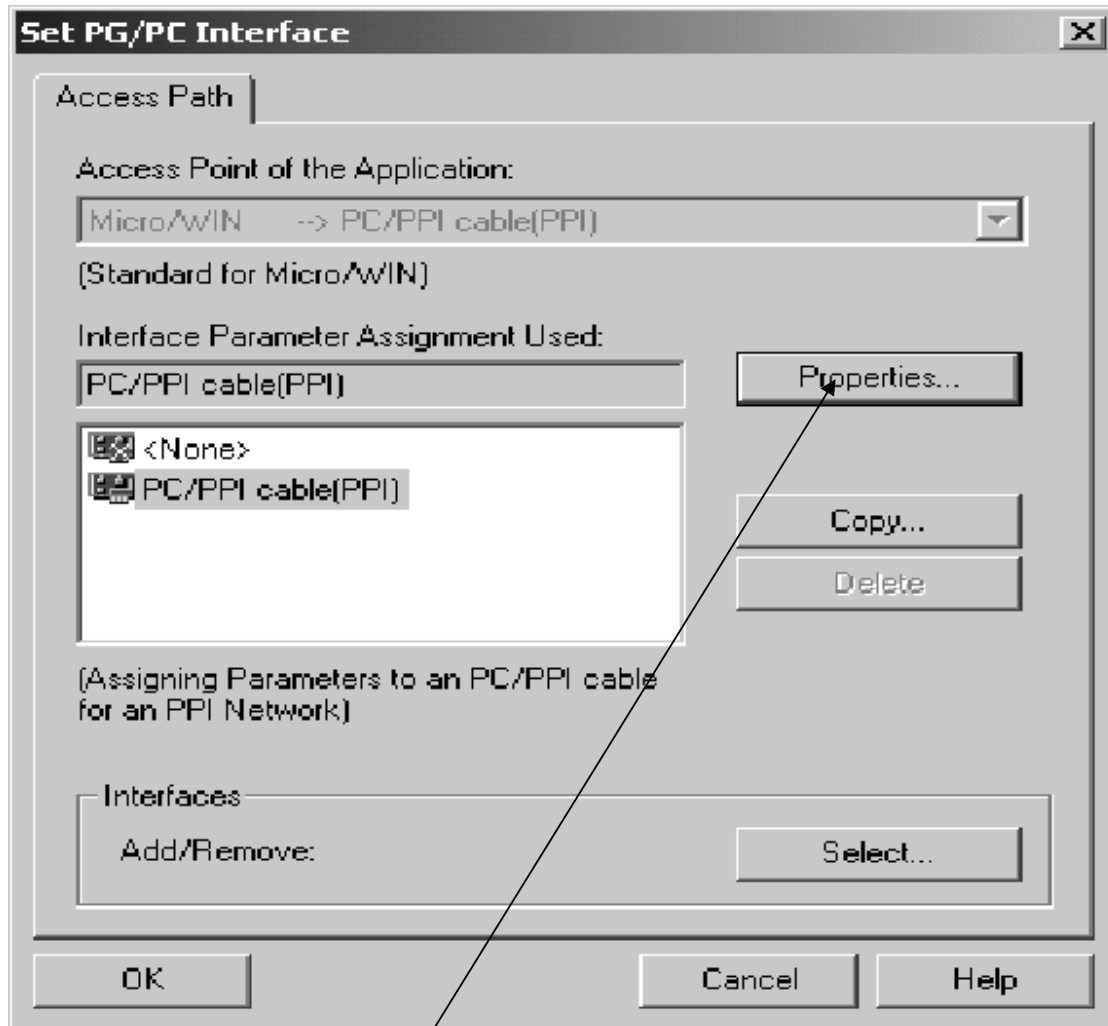


Figure 3.4 (b)

2

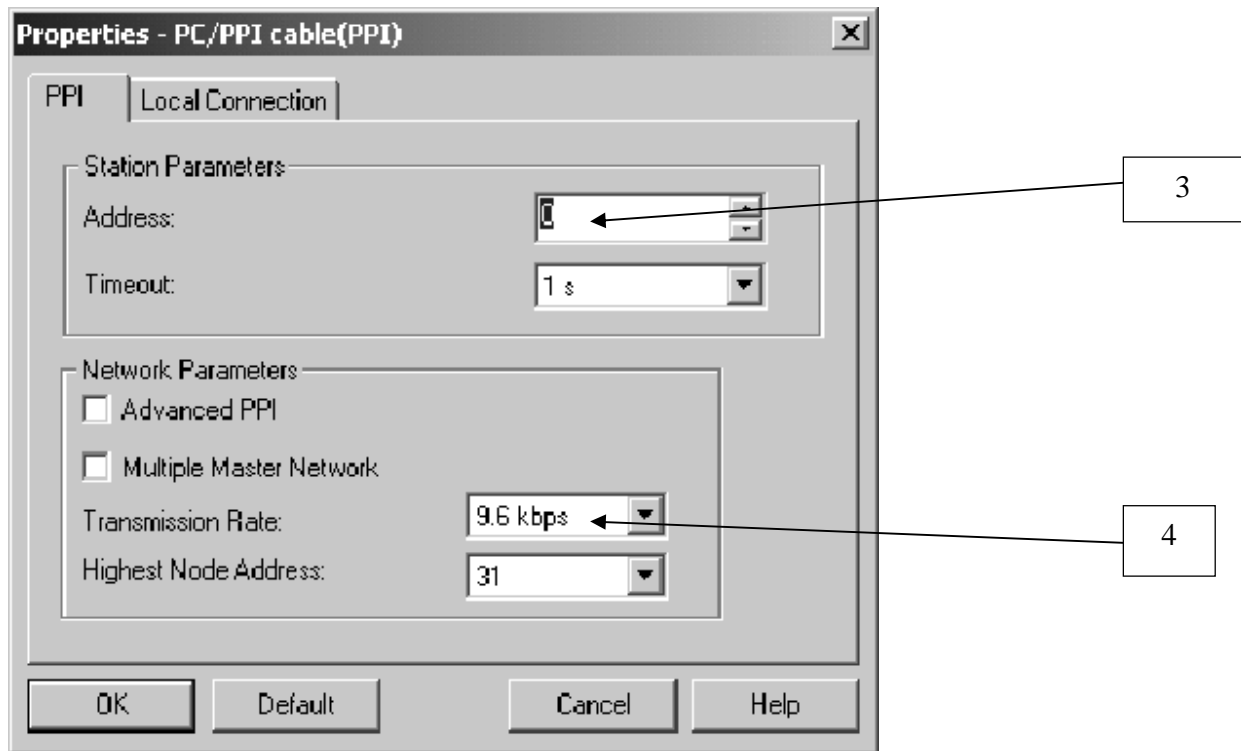


Figure 3.4 (c)

The subsection describes the setting of the Baud rate and network address for the S7200. Using the same principles, you also need to configure the baud rate and network address for S7-200. The baud rate and network address and then stored in the system block of the S7-200. Parameters for the S7-200 are selected as well as downloading the system block to the S7-200. For the default setting, the baud rate of the S7-200 is 9.6kBaud and network address of 2 is used. The baud rate and network address for the S7-200 are set using the below steps which are also describes in figure 3.4(d)

1. Select network address for S7-200
2. Select the baud rate for the S7-200
3. Download the system block for S7-200

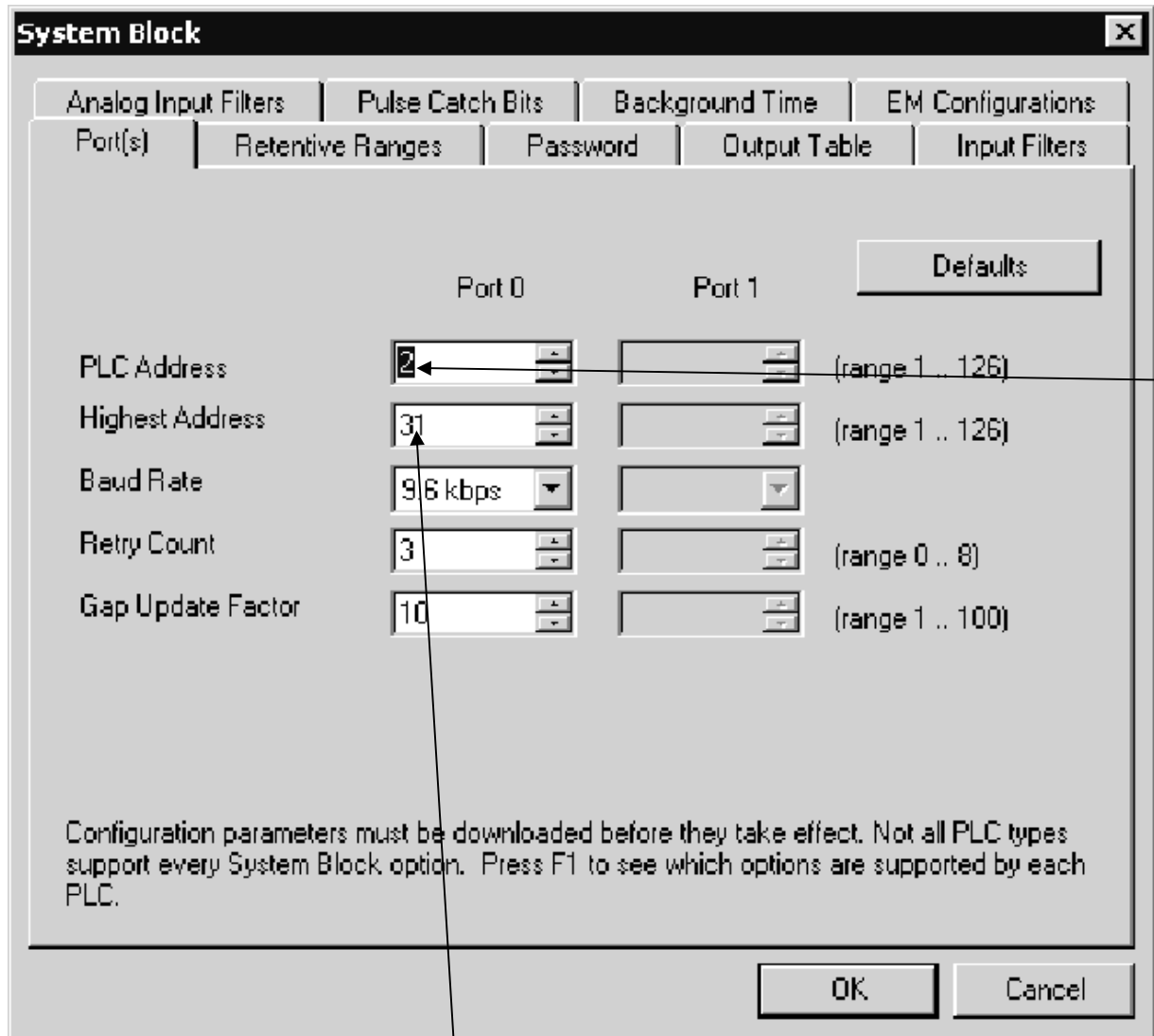


Figure 3.4 (d)

The setting of the remote address is very important for the communication in STEP-7—Micro/Win. The remote address of the S7-200 and communication (COM) port of STEP-7—Micro/WIN must be set first to match the current setting of the S7-200 as in figure 3.4(e) before downloading the updated setting to the S7-200. There are two ways of reconfiguring the COM port. One possible way is to click the

communication icon in the navigation bar or alternatively, select the View>component>Communication menu command,

1. Select the remote address
2. Ensure that the parameters for the COM port, the remote S7-200 port, and the PC/PPI cable match or else the communication fails.

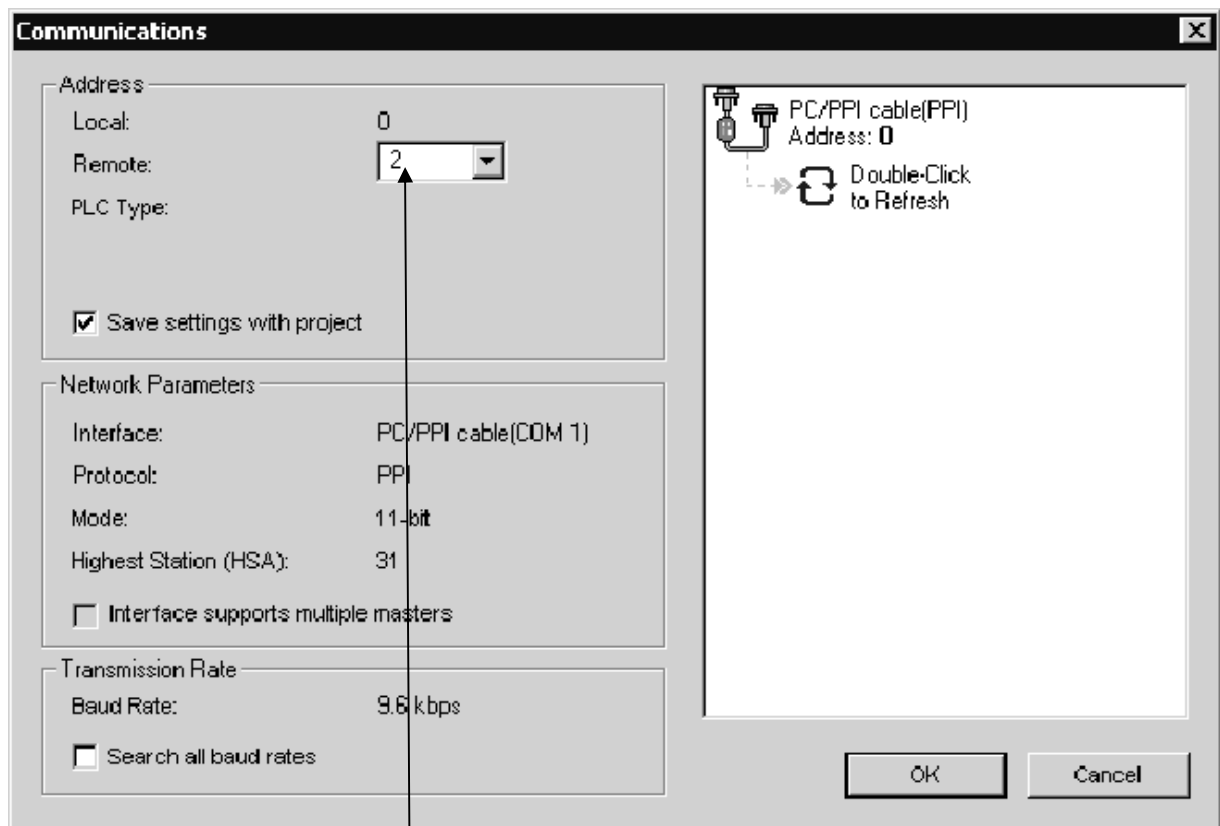


Figure 3.4 (e)

The identification for other 1s that are attached to your network can be searched as well the baud rate for the specific network. PC/PPI cable and STEP-7—Micro/WIN can only search baud rate at 9.6kBaud and 19.2kBaud. While for CP card and STEP-7—Micro/WIN, the baud

rate can be search at 9.6KBaud, 19.2kBaud and 187.5KBaud. Below steps are followed to achieved the searching of the baud rate, also describes in the figure 3.4(f)

1. Open the communication dialog box and double click the refresh icon to start the search
2. To select the all baud rates, select search all baud rates check box

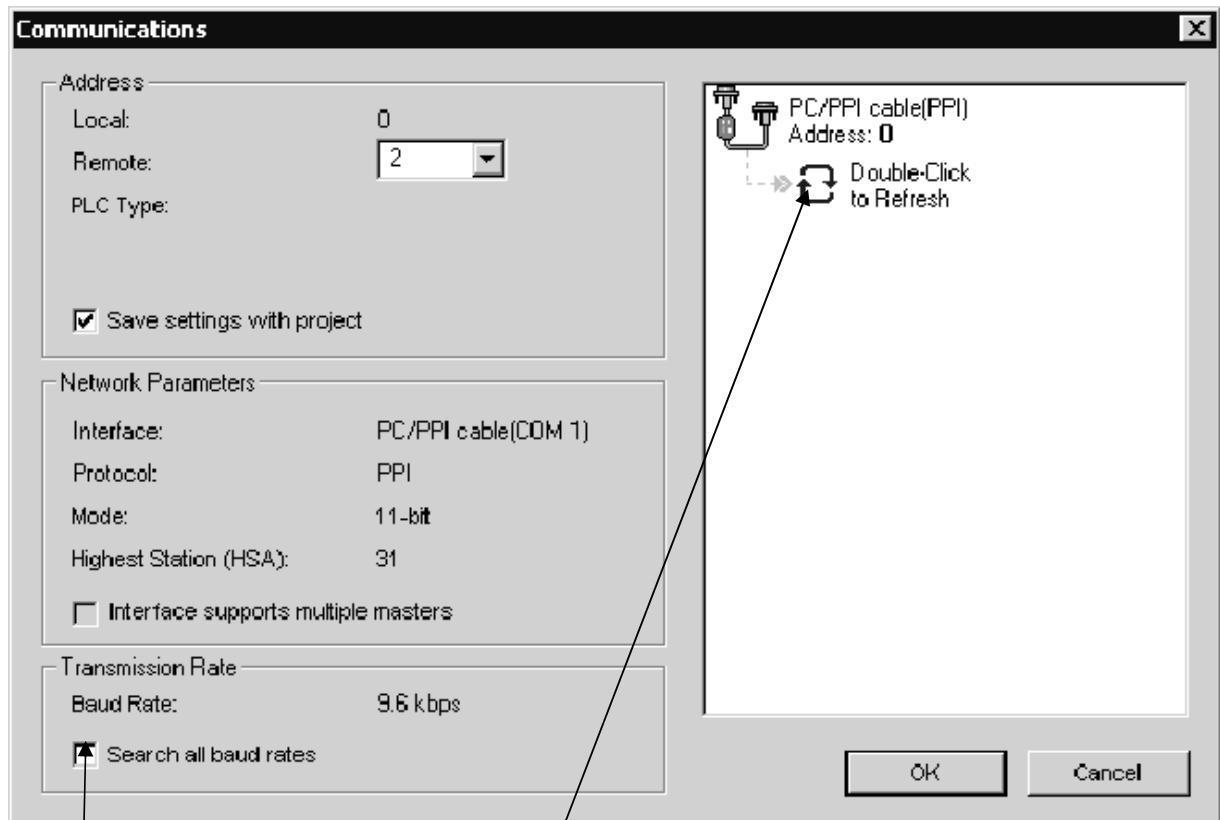


Figure 3.4 (f)

2

1

Another important stage is the selection of the communication protocol for the network. These include the Point-to-Point Interface (PPI), Multi-Point Interface (MPI) and PROFIBUS. Each of the above mentioned protocols are generally supported by the S7-200 CPUs.

PPI is considered to be master-slave protocol which is responsible of sending the request to the slave devices as well getting the response from the slave device. The slave devices do not response to messages until the master sends them a request or polls them for the response. The communication between the masters and slaves are achieved by the means of a shared connection that is using the PPI protocol. There is no limit given by PPI to the number of masters but you cannot install more than 32 masters on the network.

MPI is a more advance version which allows communication for both master-master and master-slave. The communication of S7-200 CPU and STEP-7 Micro/WIN is achieved by establishing the connection of master-slave but the MPI does not communicate with an S7-200 CPU which functions as master. There is separate connection for the network devices to communicate. There is also limited communication between the devices to the number of connections supported by the S7-200 CPU or EM277 modules. Refer to table (3.4c) for the number of connections supported by S7-200.

Module	Baud rate	Connection
S7-200 CPU Port 0	9.6kBaud, 19.2kBaud, 187.5kBaud	4
Port 1	9.6kBaud, 19.2kBaud, 187.5kBaud	4
EM277	9.6kBaud to 12Mbaud	6 per module

Table 3.4(c); Number of Connection for the S7-200 CPU and EM277 modules

Source: Source: CITECT SCADA Help

The PROFIBUS Protocol is a more advanced protocol which is designed specifically for the high speed communications with distributed I/O devices. There are several PROFIBUS devices available from many different manufacturers. They range from simple input or output modules to motor controllers and PLCs. Typically, PROFIBUS has one master and many slave input/output devices as much as possible. The configuration of the master device determines the types of I/O slaves that are connected as well as determining the addresses. The master initialises the network and ensures that there are match configurations of slave devices on the network. The output data can then be written to the slave device as well reading them from the slave device continuously by the master device.

Chapter 4

4.0 Supervisory Control and Data Acquisition

4.1 Introduction

SCADA stands for Supervisory Control and Data Acquisition. It is used to obtain information from PLC devices as well as providing an interface for controlling the system (OXenham, 2001). It is commonly used as an industrial control system such as controlling processes, remote monitoring, plant visualization and controlling the system by providing trends, alarms, time schedule and data lodging. These are displays in term of charts, reports, and text documents. This collected information can be used by the system management for tracking product and monitoring the stock usage in which they are achieved through PC by windows keyboard and mouse. It can also be used to monitor and operate the facilities through internet and hence, cost reduction. The SCADA system usually contains the following subsections namely;

- Human Machine Interface also known as HMI. HMI system supplied functions of superset to the SCADA. It is where human operator can present the process data and monitor the performance of the system as well the controlling of that process data.
- Supervisory system which collects the data on the process and sending commands to the process
- Remote Terminal Units (RTUs). This connects sensors to the process, converts the signal of the sensors into digital data and precedes that digital data to the supervisory system.
- The final part is communication which connects the supervisory system to the Remote Terminal Units.

4.1.1 SCADA hardware

A SCADA system is comprised of several numbers of Remote Terminal Units also known as RTUs which collects the data and send them back to a master station through a communication system (Bailey & Wright). This will allow the performance of the system to be monitored and controlled. Moreover, its benefits are in wide range such as efficiency, flexibility, Reliable choice of global manufacturers. Implementing SCADA will maximize the productivity, reduced the operation and maintenance costs, improve product quantity, and secure and protect investment. Bailey and Wright provide the five levels of hierarchies as below;

- Communication system
- Field level instrumentation and control devices
- Marshalling terminals and RTUs
- Master stations and,
- The commercial data processing department computer system.

To elaborate more, the interface is provided by RTU to the field digital and analogue sensors which are located at each and every remote site. Nevertheless, the pathway for communication between the master station and remote sites can then be provided by the communication system. This communication system can be in form of telephone line, fibre optic, wire, radio, microwave and possibly satellite.

4.1.2 Communication between PLC and SCADA

The SCADA is capable of communicating with lots of devices that are used to monitor and control such as Programmable Logic Controllers, Remote Terminal Units, Distributed control systems, loop controllers etc (reference?). Figure 4.1(a) provides the overview of how it can directly communicate with inputs/outputs thus; allow the transfer of data from the plant to the device as well allowing the transfer of the data from the device or supervisory control system to plant or factory (reference)

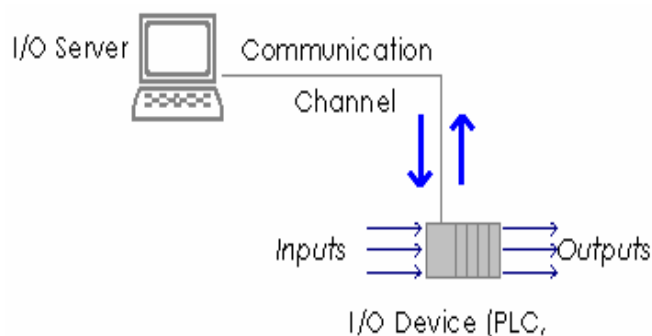


Figure 4.1(a)

Source: CITECT SCADA Help

To enable CitectHMI/SCADA to communicate with an I/O Device you need a device driver. This is the interface between CitectHMI/SCADA and the I/O Device which implements the communication protocol(s) of the I/O Device. The structure of a CitectHMI/SCADA device driver is extremely flexible. It allows for a single driver to communicate using several hardware boards, each with several hardware ports, with each port communicating with several I/O Devices (reference?).

There are many serial communication standards used to allow the communication of I/O server and I/O devices. These include RS-232C, RS-422, RS-485. Their functions will be provided below

The most common serial data communication interface standard is RS-232C. It defines the electrical as well as the mechanical details of the interface however, it does not define a protocol. The standard covers the electrical signal characteristics, the mechanical interface characteristics (pin out etc) and functional description of control signals etc (reference?).

- Point-to-point communication. Between only 2 devices.

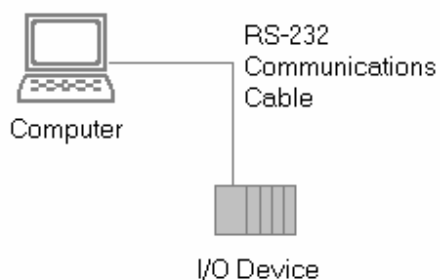


Figure 4.1 (b)

Source: CITECT SCADA Help

- Communication is full-duplex. A single wire for each direction and a ground wire. This means that generally only 3 wires need to be connected for most applications. The diagram below shows the 'standard' pins for a DB-25 Connector.

Secondly the RS-485 is an improved version of RS-422. This standard covers the electrical signal characteristics and functional description of control signals only. It does not define the protocol, but the protocol used should support multiple unit addressing and bus contention to fully utilise this standard. The major advantage is that all devices can transmit and receive on the same line.

- Electrically very similar to 422. Logic levels, transfer rates and maximum distance are almost identical.
- RS-485 supports multiple transmitters and receivers on each line. This is the improvement on RS-422.

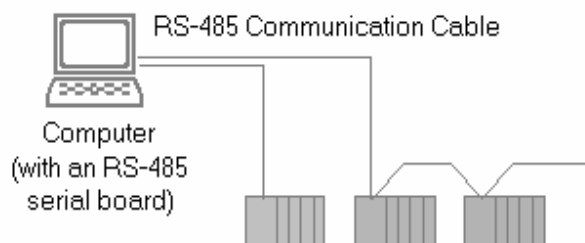


Figure 4.1 (c)

Source: CITECT SCADA Help

Communication may be either half-duplex or full-duplex. Two wires are used for each direction and also one ground wire. This means that only three wires need to be connected for most half-duplex applications. Five wires are needed for most full-duplex applications. The diagram below shows a commonly used pin arrangement for a DB-9 Connector

4.2 Features of CITECT SCADA

The Citect SCADA has a lot of features in its software. These include User interface, graphics displays, Alarms, Trends, RTU (and PLC) interface, Scalability, Access to data, Database, Networking, Fault tolerance and redundancy and Client/server distributed processing(Bailey & Wright). These features are listed in the bullet points below (Bailey & Wright).

4.2.1 User interface include

- Keyboard commands

This command has key sequences that are entered by an operator on the key and command which are executed when the key sequence is entered. The keyboard command can be used to define the operator that (Bailey & Wright);

- Only when an operator positions the cursor over a specific object on a graphics page.
- For all graphics pages displayed on the computer screen
- Only when a specific graphics page is displayed
- Only when an operator positions the cursor over a specific object on a graphics page

Other user interface includes;

- Mouse
- Trackball
- Touch screen
- Graphics Displays
- Resolution
- Customer-configuration, object oriented and bit mapping
- Unlimited number of pages

4.2.2 Alarms

The basics of Alarms are used to constantly monitor and alert the operator if there are faults in the equipment. Hardware Alarms has Citect that always runs diagnostic routines and thus, checking all peripheral equipment likes I/O devices. This allows faults to be reported automatically to the operator. It is fully integrated within Citect thus, no configuration is necessary. Considerably, alarms that report fault conditions in the plant must be configured. Some more alarms are summarized in bullet points below (_Bailey & Wright)

- Client server architecture
- Timer stamped alarms to one milliseconds precision or better
- Single network acknowledge and control alarms
- Alarms are shared to all clients
- Alarms displayed in chronological order
- Dynamic allocation of alarms pages
- User-defined format and colours
- Up to four adjustable trip points for each analogue alarm
- Deviation and rate of change monitoring for analogue alarms
- Selective display of alarms by category (256 categories)
- Historical alarm and event logging
- Context- sensitive help
- On-line alarm disable and threshold modification
- Alarm-triggered reports
- Operator comments can be attached to alarms

4.2.3 Trends

A visual representation of past and current activities improves understanding of the performance of the plant. Trend graph allows you to display the values of the variable in graphical format. As these values change over time, the graph moves across the page so that the latest values are always displayed. Scrolling back through historical data is also allowed. Historical data collection continues even when the displays are not active. Some useful functions are summarized below by Bailey & Wright as follow;

- Client server architecture
- True trend printouts not screen dumps
- Rubber band trend zooming
- Export data to DBF,CSV files
- X/Y plot capability
- Events based trends
- Pop-up trend display
- Trend gridlines or profiles
- Background trend graphics
- Real- time multi-pen trending
- Short and long term trend display
- Length of data storage and frequency of monitoring can be specified on a per-point basis
- Archiving of historical trend data
- On-line change of time-base without loss of data
- On-line retrieval of archived historical trend data
- Exact value and time can be displayed
- Trend data can be graphically represented in real time

- RTU and PLC interface
- All compatible protocols included as standard
- DDE drivers supported
- Interface also possible for RTUs, Loop controllers, bar code readers and other equipments
- Driver toolkit available
- Operates on a demand basis instead of the conventional predefined scan method
- Optimisation of block data requests to PLCs
- Rationalisation of network user data requests
- Maximisation of PLC highway bandwidth

4.2.4 Scalability

- Additional hardware can be added without replacing or modifying existing equipment
- Limited only by the PLC architecture

4.2.5 Access to data

- Direct, real-time access to data by any network user
- Third-party access to real-time data such as excel
- Network DDE
- DDE compatibility: read, write, etc
- DDE to all I/O device points

- Clipboard

4.2.6 Database

- ODBC driver support
- Direct SQL commands or high level reporting

4.2.7 Networking

Citect uses a concept called “Scalable Architecture” which means you can implement Citect on a single computer or small network and expand at a later stage without needing to change the hardware, software or system configuration.

Each Citect computer on a network is called a Display Client and at least one of those computers needs to also be set up as a Citect Server. Each Citect Server can be setup for the processing of I/O information and Alarms, Reports or Trends.

As the project size increases, the Alarms, Reports and Trend functions can be run on separate server computers.

In addition, Citect also allows you to add redundancy for each of your servers and your network connections, so that no data is lost when a computer fails or a network link is damaged. Some of them are summarised in bullet points below.

- Supports all NetBIOS compatible networks such as NetWare, LAN manage, Windows or Workgroups
- Supports protocols NetBEUI, IPX/SPX, TCP/IP and so many others
- Centralised alarm, trend, report process
- Dual networks for all full LAN redundancy
- No network configuration required
- May be enabled via single check box, no configuration

- LAN licensing is based on the number of users logged onto the network, not the number of nodes on the network
- No files server required
- Multi-user system, full communication between operators
- RAS and WAN supported with high performance
- PSTN dial up support

4.2.8 Fault tolerance and redundancy

Many industrial plants cannot afford to have their Citect monitoring system fail at any time. Building redundancy into your system can prevent the loss of control and monitoring of your Citect system. Types of redundancy are provided below;

- Dual networks for all LAN redundancy
- Redundancy can be applied to specific hardware
- Supports primary and secondary equipment configuration
- Intelligent redundancy allows secondary equipment to contribute to processing load,
- Automatic changeover and recovery
- Redundant writes to PLCs with no configuration
- Mirrored disk I/O devices
- Mirrored alarm servers
- Mirrored trend servers
- File server redundancy
- No configuration required

4.2.9 Client/ server distributed processing

- Open architecture design

- Real-time multitasking
- Client/server fully supported with no user configuration
- Distributed project updates
- Concurrent support for multiple display nodes
- Access any tag from any node
- Access any data(trend, alarm, report) from any node

Chapter 5

5.0 Conveyor Belt Control using the Siemens S7200 PLC based controller

5.1 introduction

As mentioned previously, PLC is one of the leading engineering and manufacturing software that can be used in power generation, process industries, manufacturing industries, refining and fabrication industries to mention few. Its applications and features were used to achieve controlling the operation of the conveyor subsystem.

This chapter will discuss in detail the inputs and outputs requirement in this project, the description of ladder diagram and finally the procedures that were used to achieve the operation of the conveyor belt system. Inputs and outputs will be discussed deeply including the required inputs and their role as well as the required output and their function. Ladder diagram used in this project will be discussed in detail featuring ladder of each network as well the function of that network. Finally the testing procedures will be also outlined detailing the physical demonstration as well the scenarios that were considered to be achieved.

5.2 input and output requirements

As mentioned previously, inputs and outputs are the most important functions in all PLCs. They are sources that drive and control the system. This section of the report will discuss the required inputs and outputs of this project. The required inputs of this project are up to seven (7) physical inputs (I) and five (5) physical outputs (Q). Their role in this project will be discussed in detail. The seven inputs include Top Dead Centre (TDC, I0.0), Bottom Dead Centre (BDC, I0.1), Height Indicator(I0.2), Start button(I0.5), Stop button(I0.6), first optical sensor (OPTO1,I0.3) and second sensor(OPTO2, I0.4). The outputs include Dispense Solenoid (Q0.3), Height measurement motor (Q0.2), Conveyor (Q0.4), Plastic bin (Q0.5), Metal bin (Q0.6).

Inputs

TDC (I0.0) is called in the first subroutine (SBR_0) to check if the height measurement gauge is still at the BDC. As for more clarification, this subroutine is called in case someone presses stop while the system was running the height measure. This subroutine is only called at the start.

BDC (I0.1) is also used in the second subroutine (SBR_1) to bring down the height measurement gauge to its Bottom Dead Centre for the performance of height measurement of the washer. In this subroutine, the Height Indicator (I0.2) is also energised to indicate if the height of the washer is within the specified tolerances and thus, setting the flag of the height for simplicity of material detection.

The TDC input I0.0 is also used for the third subroutine (SBR_2) to return the motorised height measurement gauge to its TDC after performance of height measurement. This is also

discussed more deeply in the section of ladder diagram featuring its ladder configuration.

The conveyor cannot start or it can run continuously if the start and stop buttons are not in co-operated. Hence, physical input I0.5 is used in this project to start the operation of the conveyor system. This input is pushed to latch the special memory flag (M30.0) which is then used throughout the whole operation of the system. Pushing stop button (I0.6) will manually halt the system by clearing the start flag.

This first optical sensor is used to detection the incoming washer to alert the motor hence, bring down the height motorised gauge to it BDC. This action is simply achieved by switching input I0.3 (OPTO1) hence, setting RUN_ON_LATCH flag which will play role in starting and stopping of the conveyor belt. This flag will also energised timers (T100 and T36) to position the washer at the right place, right time and allow enough time to perform the height measurement.

The second sensor uses physical input (I0.4) to detect the type of material. This process is accomplished simply by switching I0.4 thus, setting flag AT_OPTO_2 that will be used very often during the detection of the type of the washer either metal or plastic. This sub-section is discussed more in the section of ladder diagram.

Outputs

Dispense solenoid (Q0.3) is energised for 2 seconds using T55 timer to dispense washer onto a moving conveyor belt. This output energised the electromagnetic actuator to pull armature thus, allow one washer to be pushed down. Also when the washer reaches the second optical sensor, the next washer is then dispensed using REDISPENSER flag and the operation continues until the system is halted by pushing the stop button..

Q0.2 is physical output used for the operation of height measurement. This operation includes the checking if the motorised height measurement is at its uppermost, placing down the motorised height measurement at BDC and returning motorised height measurement to its TDC. Checking is achieved by calling the subroutine SBR_0 as discussed previously as well calling subroutine SBR_1 to place down the motorised height measurement gauge. The measurement gauge can then be returned by calling subroutine SBR_2). This operation uses the concept of Boolean algebra of logic operation (OR) in which the output will be one if one of the above subroutines is on. Refer to the below diagram for more information.

The starting and stopping of the conveyor belt is controlled by physical output Q0.4. The operation of this output needs more considerations. The fact that the conveyor can get damaged if the height measure is on its motion, thus, this degradation can be avoided by stopping the conveyor (Q0.4) if the output Q0.2 is operating. RUN_ON_LATCH which uses timers (T100 and T36) is also considered in which it allows the positioning of the washer as well allows enough time for height measurement. The conveyor is also stopped when it reaches to the second sensor of material detection. This is achieved using the combination of the RESTART_CONVEYOR timer and HT_OK_AT_OPTO2 thus, allowing specific time for the detection of the type of the washer. Refer to the ladder diagram below.

Directing plastic washer into its respected bin is controlled by the physical output Q0.5. This is achieved by setting height flag in conjunction with T97 FLIPPER_2_off timer to allow 12 seconds for the washer to be placed in the plastic bin. Refer to the below ladder diagram.

Physical input Q0.2 is used to control the direction of the metal washer into its metal bin. If the

height measurement was within the specified tolerance and the metal flag is set, T33(FLIPPER_1_off) can then be used to allow 15 seconds for the metal washer to be successfully placed into its respected bin. Refer to the diagram below for it configuration.

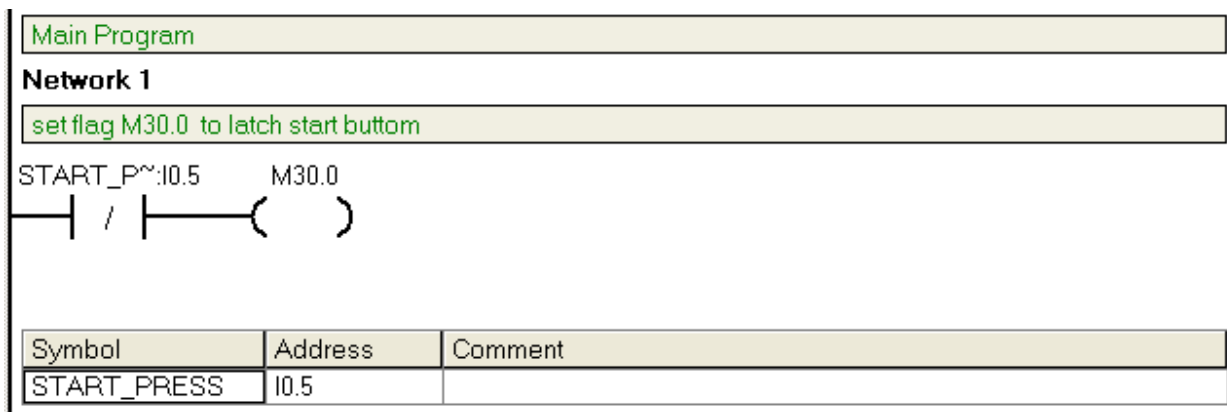
5.3 Ladder Diagram

This section describes in details the function ladder diagram used in this project. Each network is discussed in details as well as its role in the conveyor subsystem.

5.3.1 Main Program

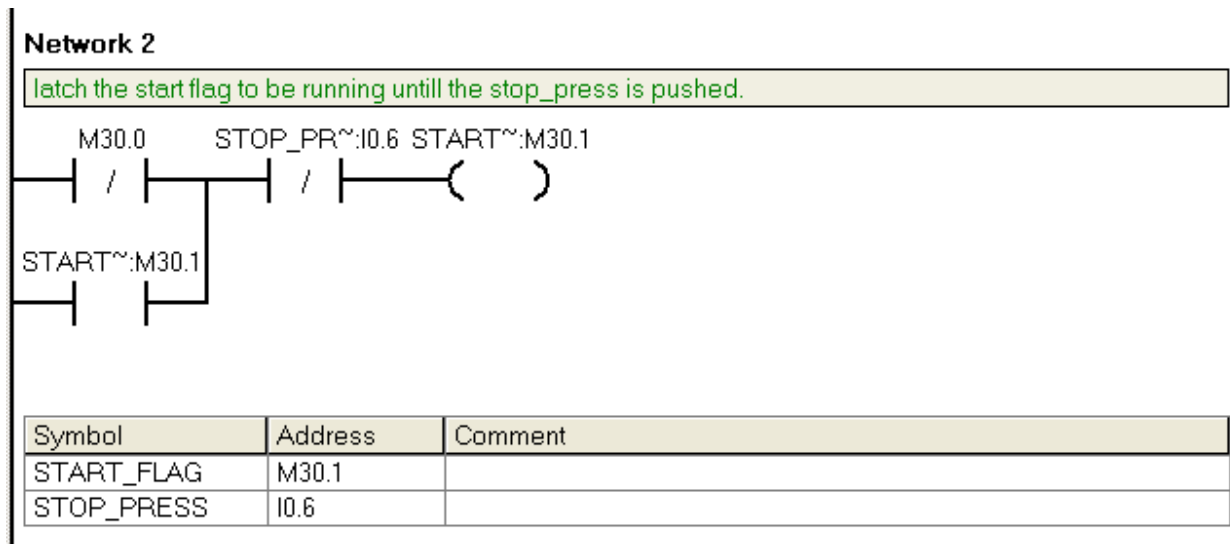
Network 1

This network energised subsystems of the conveyor by pressing the START_PRESS (I0.5) button to set target M30.0. In other words, at the start, the dispense solenoid will be activated, checking if the height motorised gauge is at TDC(TOP DEAD CENTRE) as well as thr conveyor belt which should only work if the height is at it uppermost.



Network 2

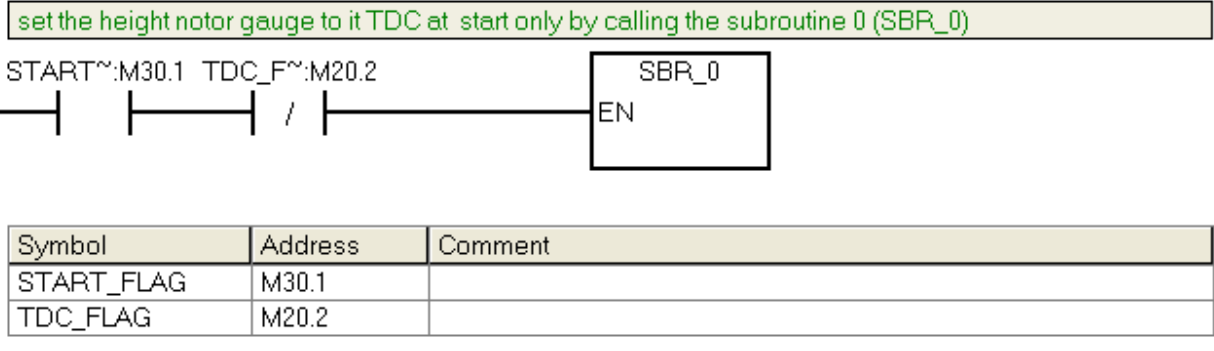
As mentioned previous, START_PRESS starts up the conveyor subsystems and should be running unless the STOP_PRESS (I0.6) is press to the operation of subsystems. This is achieved by latching the network and set the START_FLAG (M30.1).



Network 3

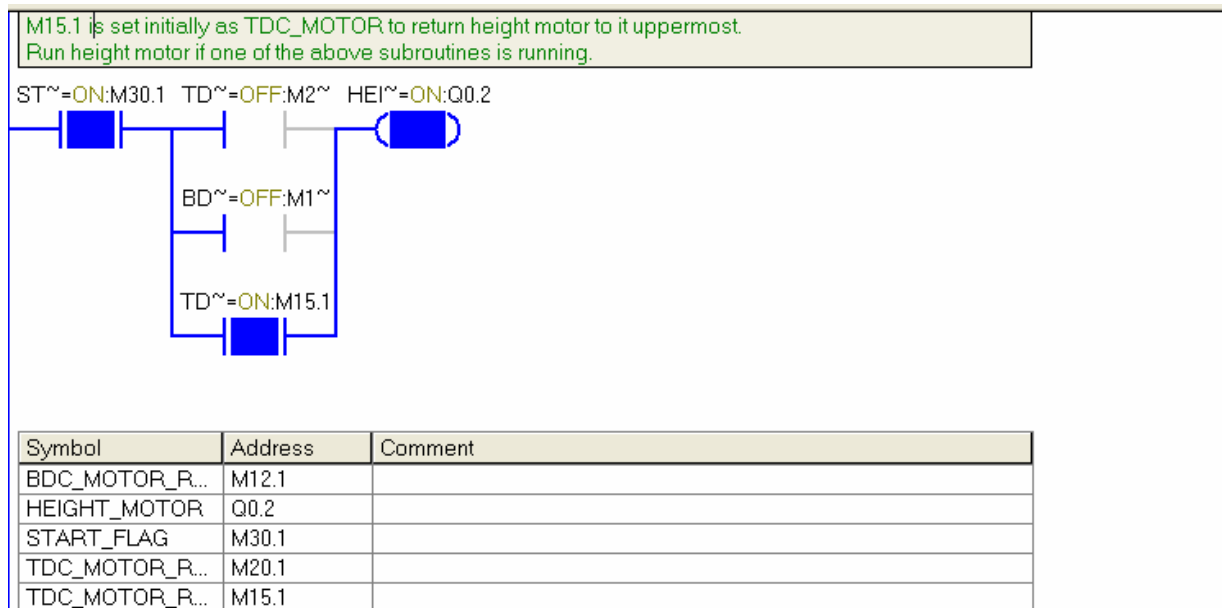
This network runs only at the starting operation of the subsystems. The START_FLAG remains latching until the stop button is pushed. The height motorised gauge is checked by calling the subroutine 0 (SBR_0) only if the TDC_FLAG which is set by SBR_0 is not on (pulse). Hence, the aim of this network is to raise the height measurement gauge to it TDC if it happened to be at it BDC (BUTTOM DEAD CENTRE) so that the height motorised arm can be able to touch and damage the conveyor belt. At the completion of running this network, flag M20.2 (TDC_FLAG) is set to block any other operation unless the stop button is pressed and hence stop the operation of the whole subsystem.

Network 3



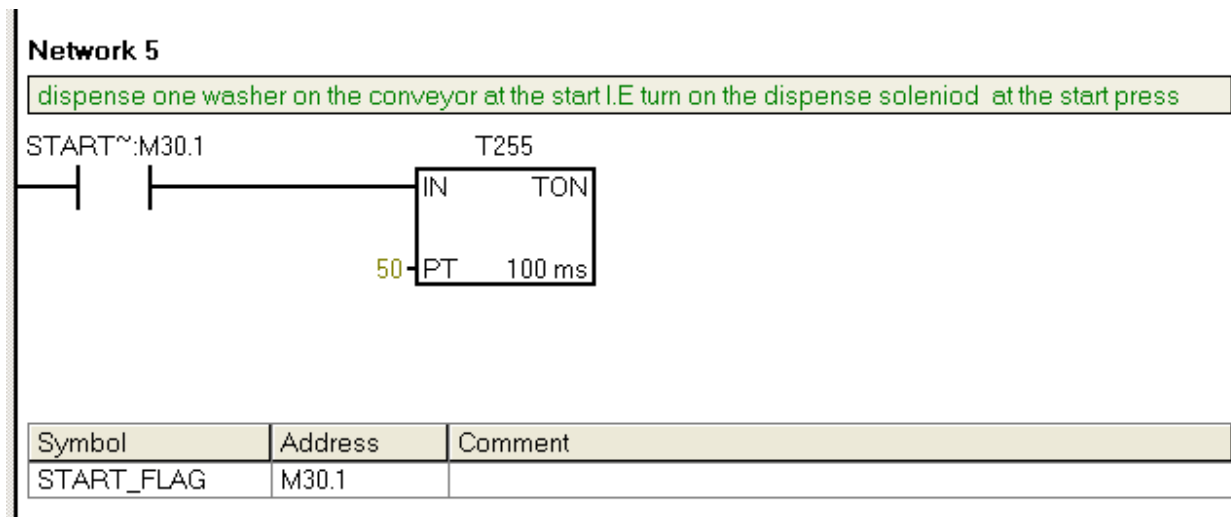
Network 4

This network is used to operate the height motorised gauge by placing either TDC, lower it to BDC and return it to TDC as respectively required. Raising it up to TDC is achieved by using subroutine 0 (SBR_0) and lowered by subroutine 1 (SBR_0) and finally, it is returned to it uppermost by subroutine 2 (SBR_0) since SBR_0 only operates at start. Therefore, the network configured if TDC_MOTOR_RUN (set by SBR_0) is on, it should raise the height motorised gauge to its TDC or if BDC_MOTOR_RUN (set by SBR_1) is on, it should lower height gauge motor to its BDC or if TDC_MOTOR (set by SBR_2) is on, it returns the height gauge motor to it uppermost. Thus, this network deals with the operation of the height measurement.



Network 5

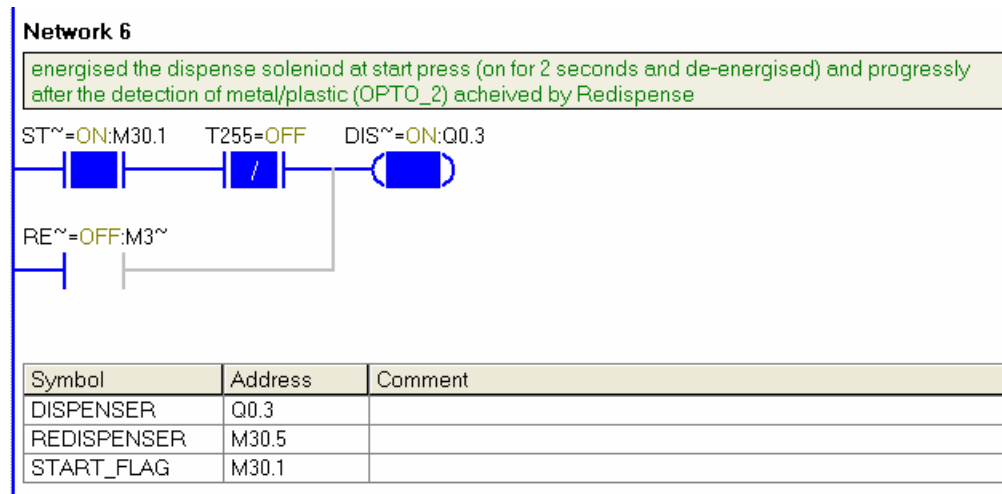
Initially at the start, the incoming washer is pushed down on to a moving conveyor belt (that is when the start button is pressed). After the washer reaches the detection operation (that is, the second sensor), another incoming washer is then dispensed onto the conveyor belt and the operation continues until the system is halted (by pressing stop button). The system must be energised and hence the T255 timer is required to time out the operation of dispensing the incoming washer and allows the next sequence of events in the subsystem.



Network 6

As described previously in network 5, the T255 timer limits the operation of the dispense

solenoid to be energised for about 2 seconds and then return to its un-energised position. REDISPENSER flag (M30.5-REDISPENSER) is set to allow the dispensation of the next washer as the first washer reaches the second sensor (inductive sensor location). The functionality involved in this operation is discussed at network 17.

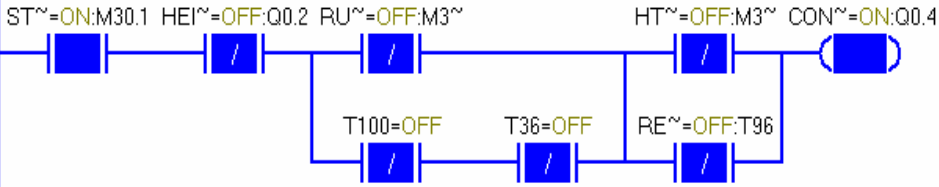


Network 7

The purpose of this network is to control the operation of the starting and stopping of the conveyor belt. To avoid damaging the conveyor belt, the energising motion is halted while the height measurement is in operation. Thus, the conveyor belt can only run if the height motor is off (height motorised gauge at its uppermost). This stopping and starting of the conveyor belt requires other operations to be considered to avoid any conflict and to successfully run accordingly. To successfully perform height measurement, T100, T36 timers or RUN_ON_LATCH are needed to correctly position the washer at the right position, at the right time and allow enough time to perform height measurement operation. When the washer reaches the second optic sensor, the combination of the RESTART_CONVEYOR and HT_OK_AT_OPTO2 is used to stop the conveyor and allow enough time to perform the detection of metal and plastic washers and restart the conveyor after the operation.

Network 7

at start, energised the conveyor and stop the conveyor for period of time to make height measurement and if the height is ok. Also conveyor is to be stop at OPTO_2 for plastic and metal detection.



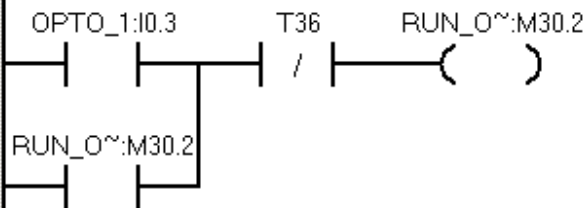
Symbol	Address	Comment
CONVEYOR	Q0.4	
HEIGHT_MOTOR	Q0.2	
HT_OK_AT_OPT...	M30.3	
RESTRICT_CON...	T30	

Network 8

The operation of this network is discussed in network 7 and will be also discuss later in network 9 and 10 respectively. It purpose to keep the conveyor belt running when it flags are set as the washer passes the first optical sensor.

Network 8

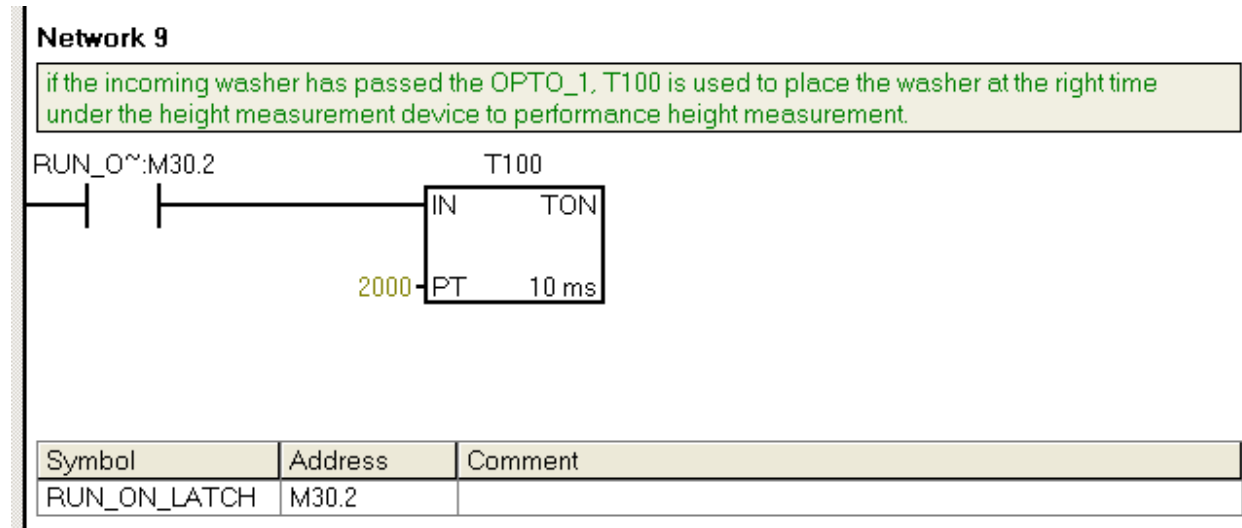
first optical (OPTO_1) is latch to detect the incoming washer



Symbol	Address	Comment
OPTO_1	I0.3	
RUN_ON_LATCH	M30.2	

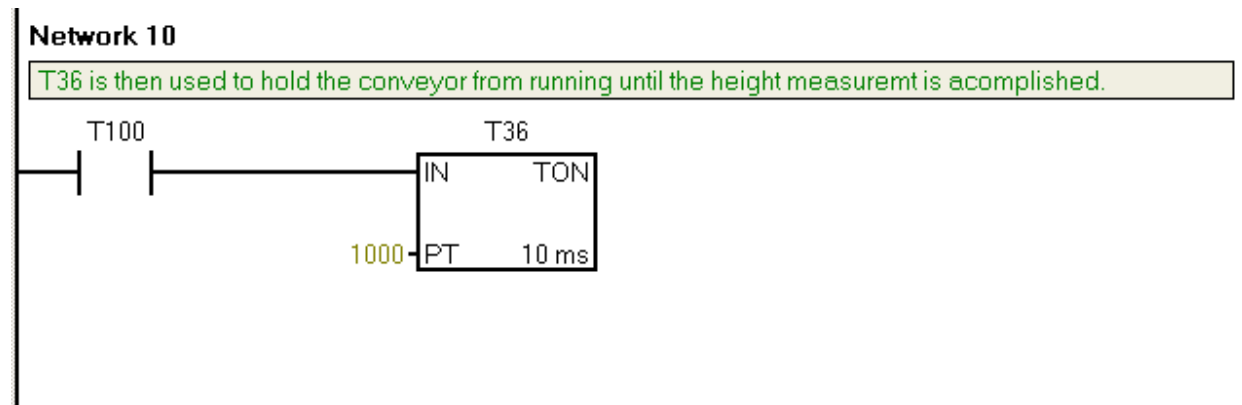
Network 9

If the incoming washer has passed the OPTO_1, T100 is used to stop the conveyor and place the washer at the right time under the height measurement device to performance height measurement, thus setting the flag of the RUN_ON_LATCH.



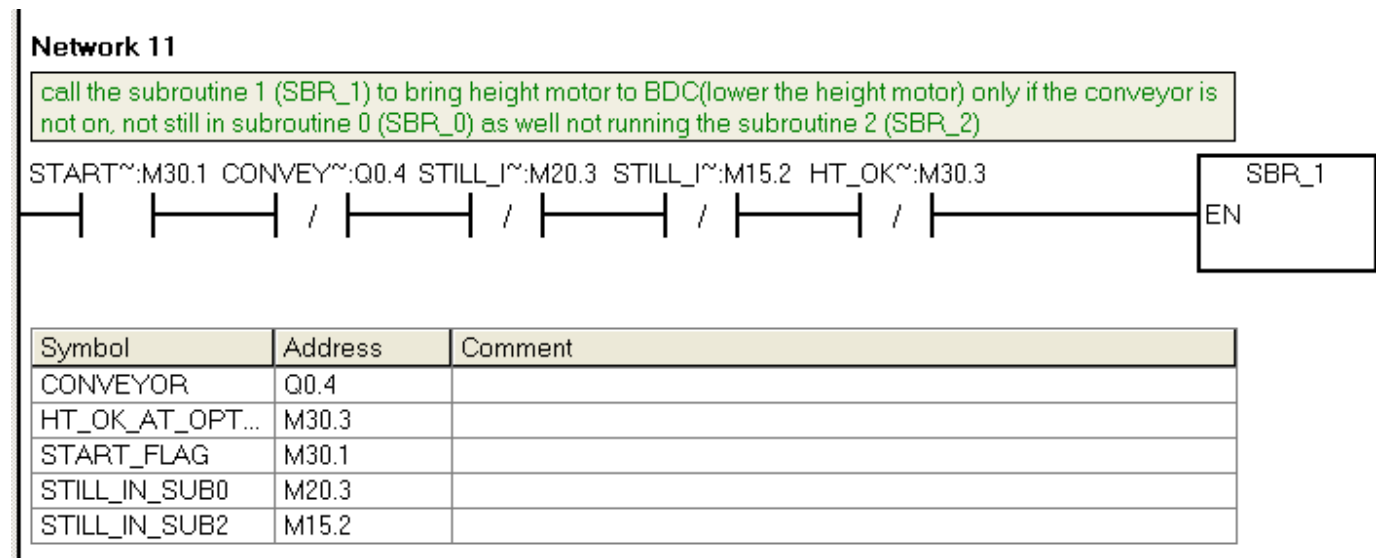
Network 10

This network uses T36 timer to hold the conveyor stops to allow the height measurement period for about 2 seconds. After 2 seconds, the RUN_ON_LATCH is cleared and hence allowing the conveyor belt to move again.



Network 11

This network calls the second subroutine SBR_1 (lowering the height gauge motor). This process is achieved if the conveyor has stopped while subroutine 0 (SBR_0) and subroutine 2(SBR_2) are not operating. Nevertheless, this subroutine will not be called again if the washer has passed the second optic sensor simply because no more height measurement is required and vice-versa for the next washer to be called.

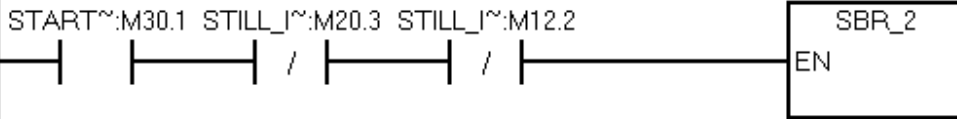


Network 12

This network calls the subroutine 2 (SBR_2) to return the height gauge motor to its upper case (TDC). This progress is truly provided that the subroutine 0 and 1 are not operating. This subroutine 2 can operate the height motor only if the TDC_COUNTER being reset (C3), performed in the subroutine 1(SBR_1) using a normally ON special memory SM0.0

Network 12

after the completion of height measurement(SBR_1), return the height motor to its uppermost not to interfere with the conveyor. this achieved by call subroutine 2 (SBR_2) only if it is not still in SBR_0 or SBR_1.



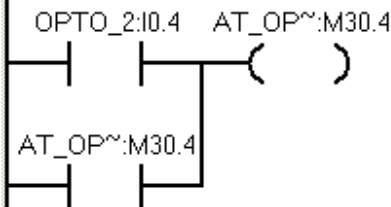
Symbol	Address	Comment
START_FLAG	M30.1	
STILL_IN_SUB0	M20.3	
STILL_IN_SUB1	M12.2	

Network 13

This network detects the incoming washer as it passes the second optical (OPTO_2) and sets its target (Latching) (AT_OPTO_2).

Network 13

detect the incoming washer at the second optical (OPTO_2) and set its target (AT_OPTO_2)



Symbol	Address	Comment
AT_OPTO_2	M30.4	
OPTO_2	I0.4	

Network 14

If the AT_OPTO_2 is high and the height measurement is ok (height within the specified range) then stop the conveyor to perform metal/plastic detection for about a two second period which is

timed out by T96. This sets HGHT_Ok flag if these conditions are satisfied and stops the conveyor for the period of 2 seconds.

Network 14

if the AT_OPTO_2 is high and the height measurement was ok(within the specified range) then stop the conveyor to perform metal/plastic detection for about a period of two second which is timed out by T96

AT_OP~:M30.4 HGHT_~:M12.3 HT_OK~:M30.3



Symbol	Address	Comment
AT_OPTO_2	M30.4	
HGHT_OK	M12.3	
HT_OK_AT_OPT...	M30.3	

Network 15

The detection of the material type, either metal or plastic is achieved by holding the conveyor for two seconds to perform the detection operation. Hence, after this operation, the conveyor belt will restart.

Network 15

restart the conveyor if the detection is done

HT_OK~:M30.3

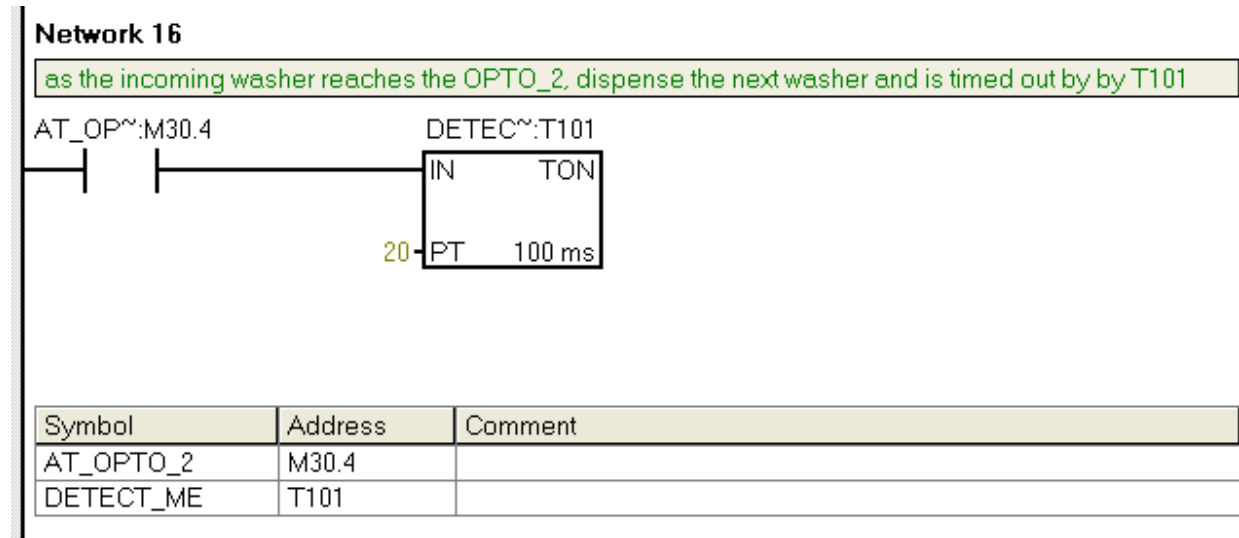
RESTAR~:T96



Symbol	Address	Comment
HT_OK_AT_OPT...	M30.3	
RESTART_CON...	T96	

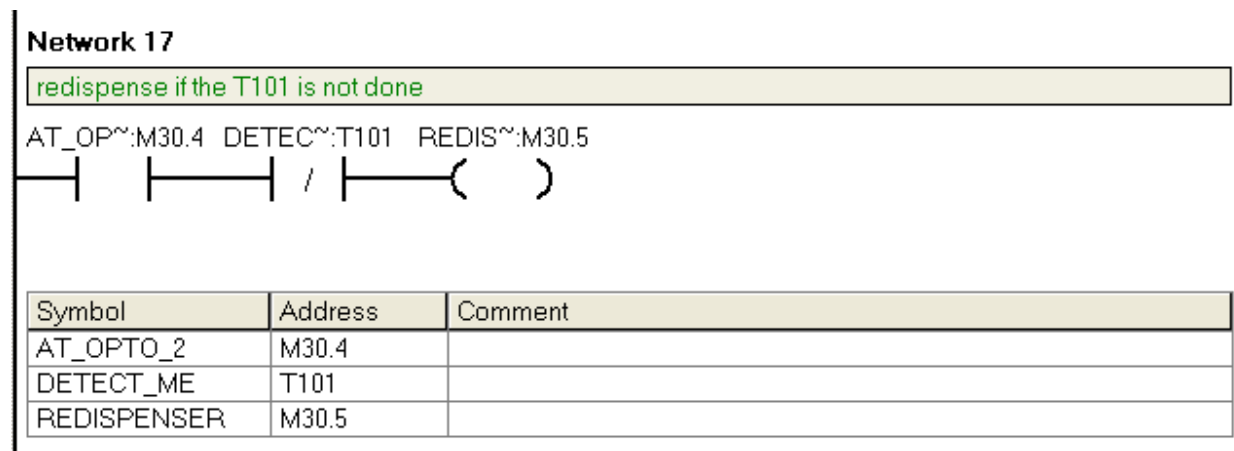
Network 16

As mentioned previously in network 6, the next washer is dispensed and timed out by T101 as the incoming washer reaches the OPTO_2. Refer to the next network 17.



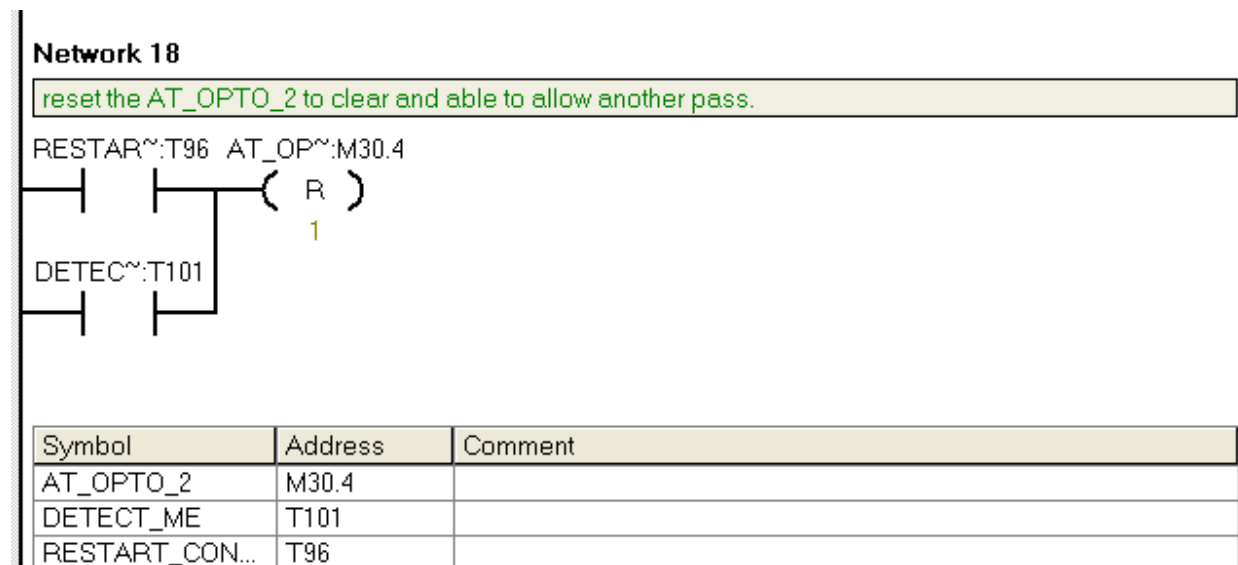
Network 17

This network process the operation of dispensing the next washer onto a moving conveyor as the first washer reaches the second optic sensor. This process is accomplished using T101 timer and its operations describes in network 16 and network 13. This process should allow the dispense solenoid to be ON for about 2 seconds.



Network 18

After the above operation, the flag M30.4 (AT_OPTO_2) is then cleared and thus, unlatching the HT_OK_AT_OPTO_2. This will allow the conveyor to run again. This process is achieved using the T96 (RESTART) and T101(DETECT_ME) timers either if the conveyor has being restart and the new incoming washer has being pushed onto the moving conveyor belt.

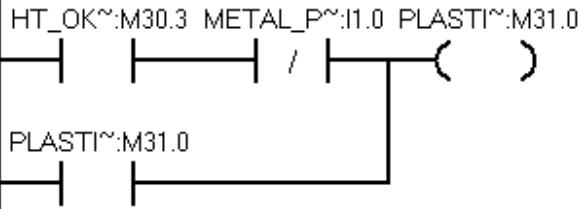


Network 19

If height measurement is ok (HT_OK) then the detection of material type either metal or plastic can be performed and set the plastic flag if plastic.

Network 19

if height measurement was ok (HT_OK) then make metal/plastic detection and set plastic flag if plastic



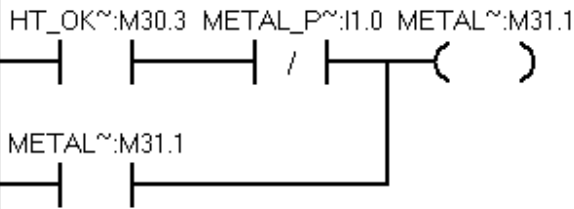
Symbol	Address	Comment
HT_OK_AT_OPT...	M30.3	
METAL_PLASTIC...	I1.0	
PLASTIC_BAG	M31.0	

Network 20

This network sets the metal flag provided that the height measurement gauge was ok and hence, taking plastic / metal detection. This process is done as the conveyor starts moving which is achieved by T96 (RESTART) timer. Otherwise, if the height measurement was gauge not to be within right range, then there is no need to do more material type detection but rather, the washer is directed to the dumping bin.

Network 20

if the height measurement was ok, take plastic/metal and set metal flag if metal



Symbol	Address	Comment
HT_OK_AT_OPT...	M30.3	
METAL_BAG	M31.1	
METAL_PLASTIC...	I1.0	

Network 21

If the height is ok and the plastic flag has being set, T97 (FLIPPER_2_off) is used for directing the plastic washer to it respected bin.

Network 21

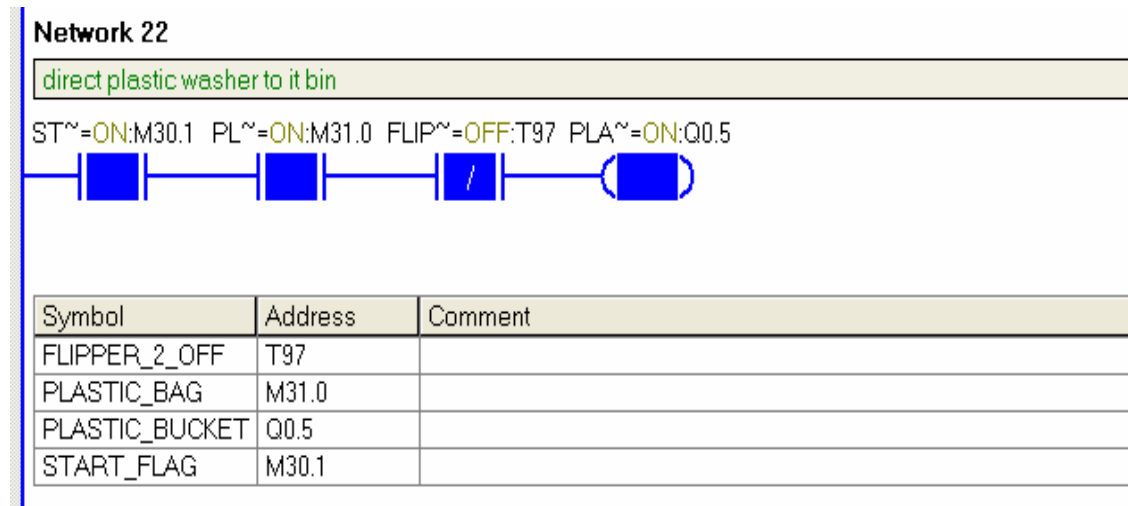
if the height ok and plastic flag has being set, T97 (FLIPPER_2_off) is used for directing the plastic washer to it respected bin



Symbol	Address	Comment
FLIPPER_2_OFF	T97	
PLASTIC_BAG	M31.0	

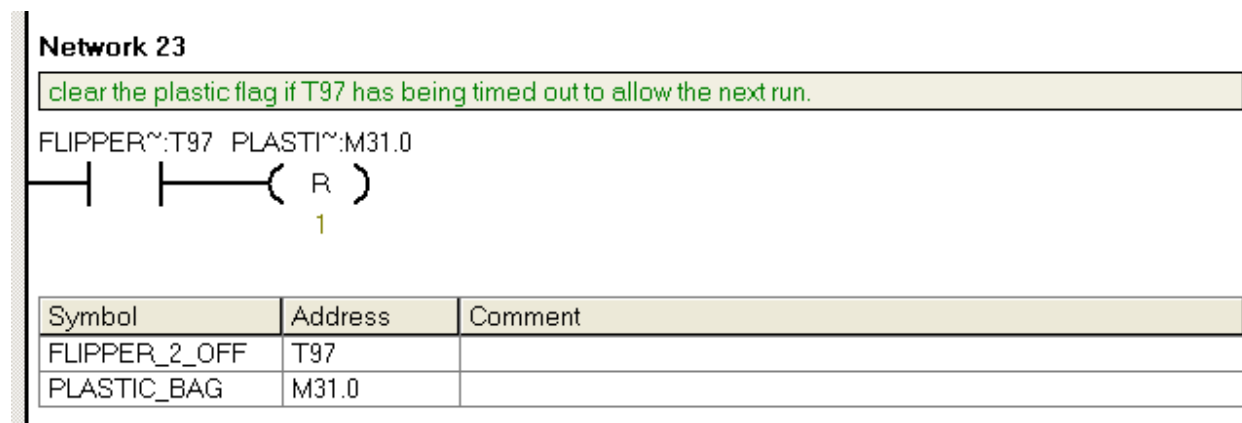
Network 22

This network operates together with network 21 to allow the duration of sorting the washer which is achieved by T97 timer. The plastic washer can be directed to its bin using the following network.



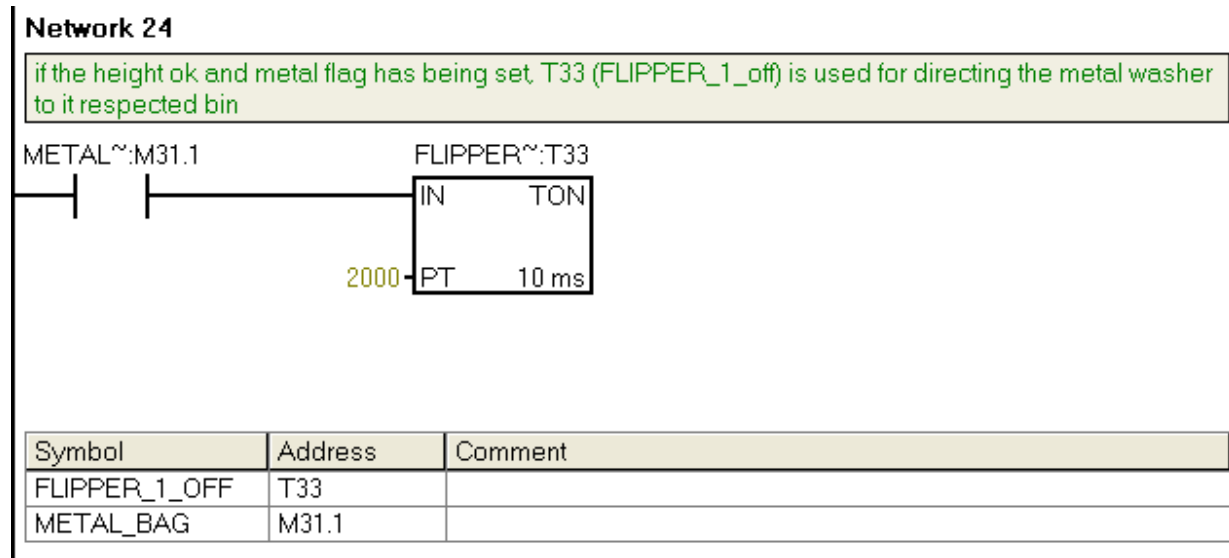
Network 23

Prior to the operation of the plastic detection, the PLASTIC_FLAG can be then cleared so that it can allow the system to set another flag for the next incoming washer.



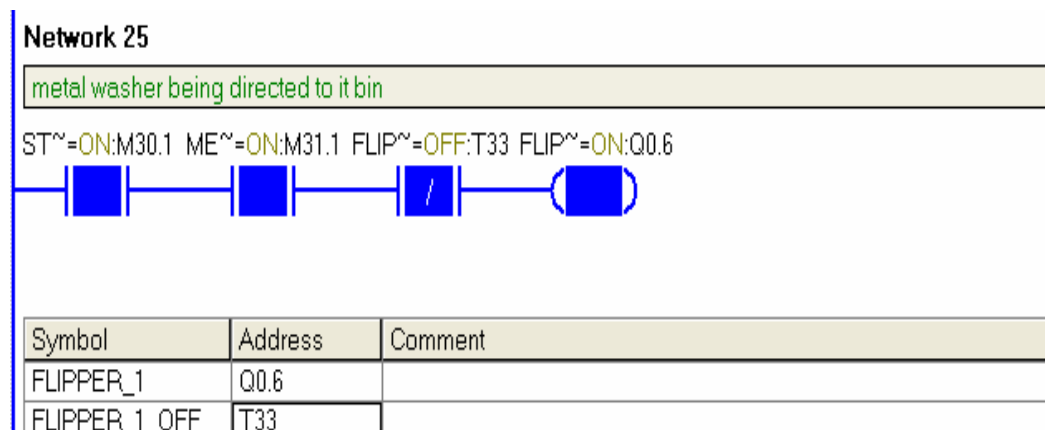
Network 24

If the height measurement gauge is ok and metal flag has being set, T33 (FLIPPER_1_off) is used for directing the metal washer to it respected bin.



Network 25

This network operates together with network 22 to allow the duration of sorting the washer which is achieved by T33 timer. This will direct the metal washer to it bin.



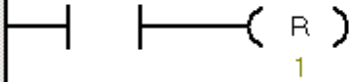
Network 26

The Plastic flag can be then cleared if T33 has been timed out to allow the next run.

Network 26

clear the plastic flag if T33 has being timed out to allow the next run

FLIPPER~:T33 METAL~:M31.1



Symbol	Address	Comment
FLIPPER_1_OFF	T33	
METAL_BAG	M31.1	

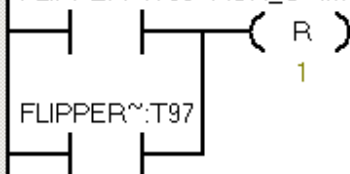
Network 27

Clear all the flags (targets) used at the end of each and every washer to avoid retaining of plastic, metal and height ok flags

Network 27

clear all the flags (targets) used at the end of each and every washer to allow clear operation accordingly.

FLIPPER~:T33 RUN_O~:M30.2

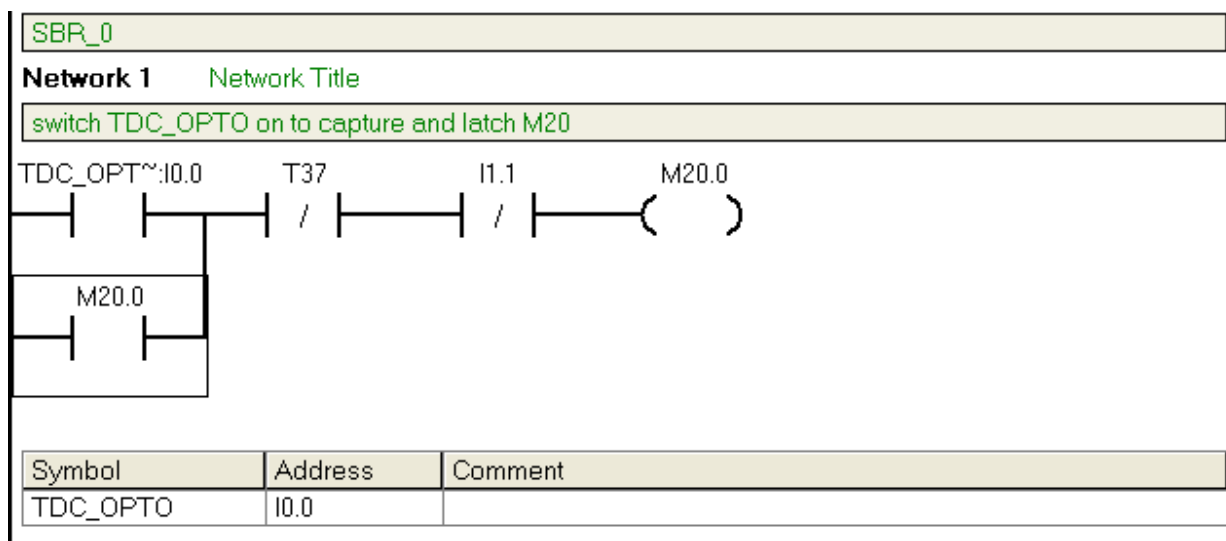


Symbol	Address	Comment
FLIPPER_1_OFF	T33	
FLIPPER_2_OFF	T97	
RUN_ON_LATCH	M30.2	

5.3.2 Subroutine 0 (SBR_0)

Network 1

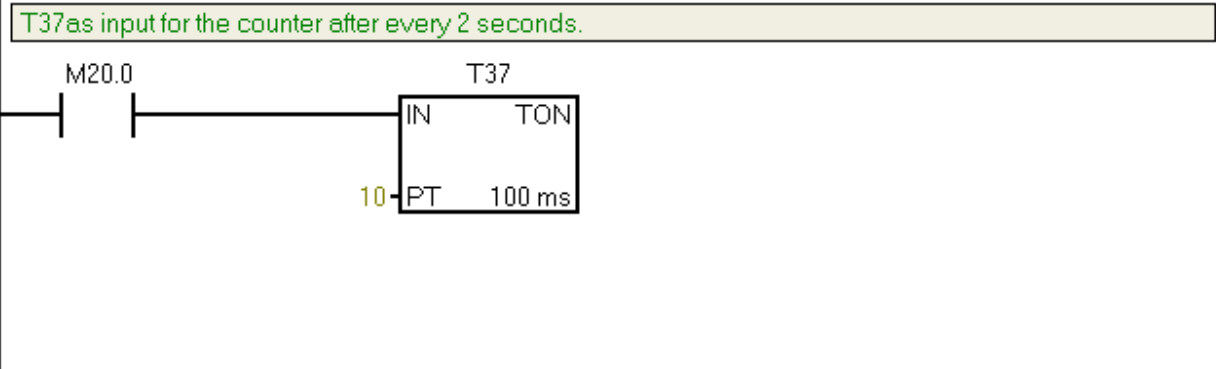
This network captures and latches the M20.0 to ON PULSE, thus indicating if the height gauge motor reaches the TOP DEAD CENTRE and other operations involved will be discussed in the next networks.



Network 2

T37 is used to reset counter 1 (TDC_COUNTER) in about 1 second. Since this subroutine is run only at the start, it should check if height motor is at BDC and take it to TDC if this condition is true.

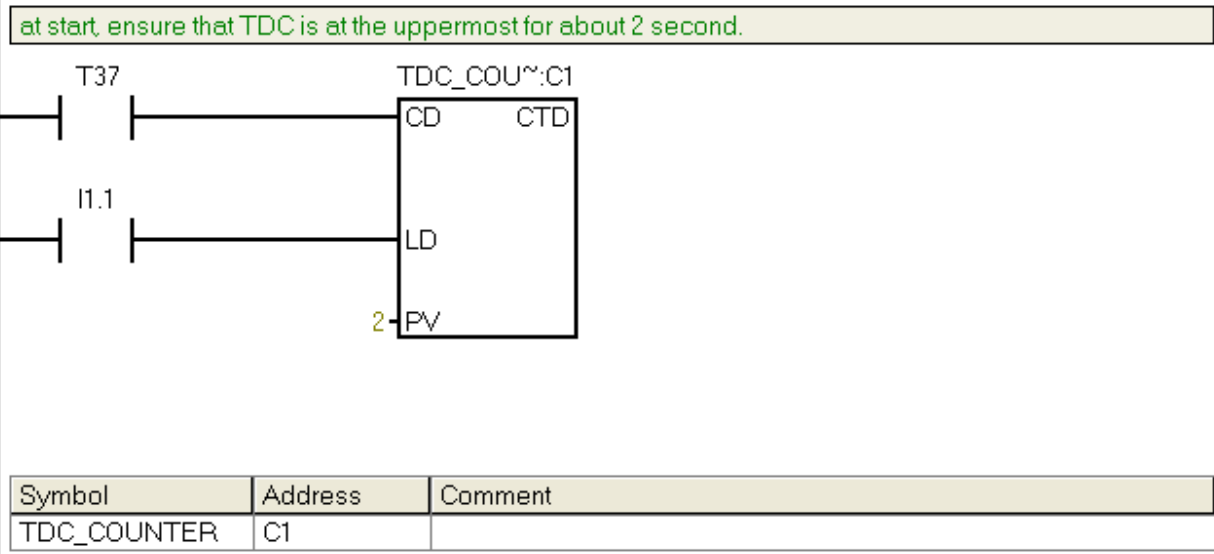
Network 2



Network 3

This network operates in conjunction with network 3 to take the height gauge to the TDC.

Network 3

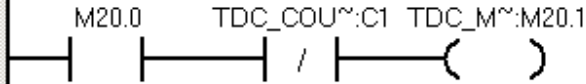


Network 4

Run the TDC_MOTOR height Motor if TDC counter is not finish otherwise, switch it off. Hence this sets M20.1 target which is only ran at the start.

Network 4

run the TDC_MOTOR height motorised gauged if TDC counter is not finish otherwise, switch it off. set M20.1 target which is run only at the start.



Symbol	Address	Comment
TDC_COUNTER	C1	
TDC_MOTOR_R...	M20.1	

Network 5

When the height motorised gauge reaches the uppermost, set the M20.2 target to stop this subroutine from running unless the stop button at the start has been pressed. This process is accomplished only if the flag M20.2 has been set. This also discussed in network 3 of the main.

Network 5

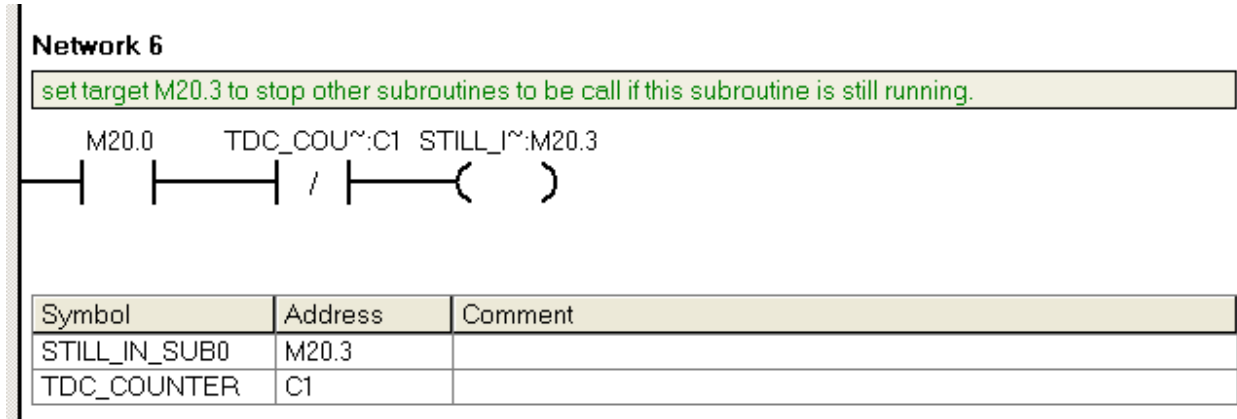
when it reaches the uppermost, set the M20.2 target to stop this subroutine from running unless the the stop button the start have being press.



Symbol	Address	Comment
TDC_COUNTER	C1	
TDC_FLAG	M20.2	

Network 6

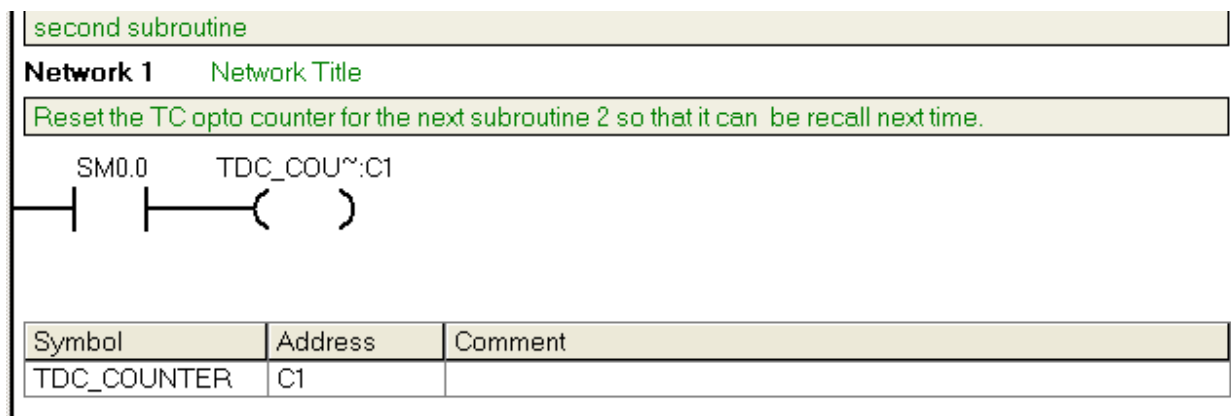
The flag M20.3 is set to stop other subroutines to be running if this subroutine is still running.



5.3.3 Subroutine 1 (SBR_1)

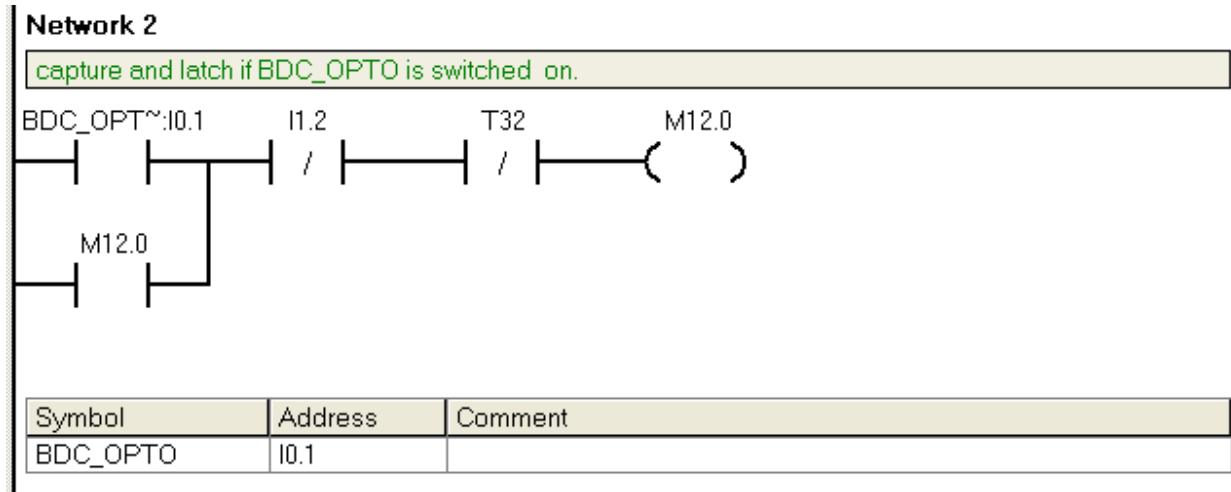
Network 1

This network reset TDC_COUNTER (C3) which was used in the subroutine 2 in order to allow the clear operation if subroutine 2 is recalled. This contributes to the fact that after each and every operation of the subroutine 1 (SBR_1), Subroutine 2 (SBR_2) is also calls to return the height gauge motor to the TDC. Therefore, C3 is reset using the normally ON special SM0.0 to successfully allow this operation.



Network 2

If the BDC_OPTO is being switched to an ON pulse, then capture and latch and set flag M12.0 which will indicate either the height motorised gauge has reached the BDC. Other operations involved will be discussed in the following networks.

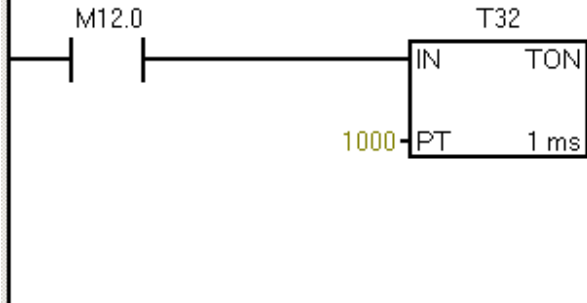


Network 3

T32 is used to reset counter 2 (BDC_COUNTER) in about 1 second. When operated the height motor is at TDC.

Network 3

T32 as input for the counter after every 2 seconds.

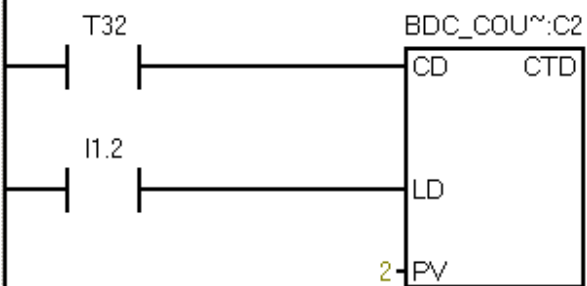


Network 4

This network operates in conjunction with network 3 to lower the height gauge to the BDC.

Network 4

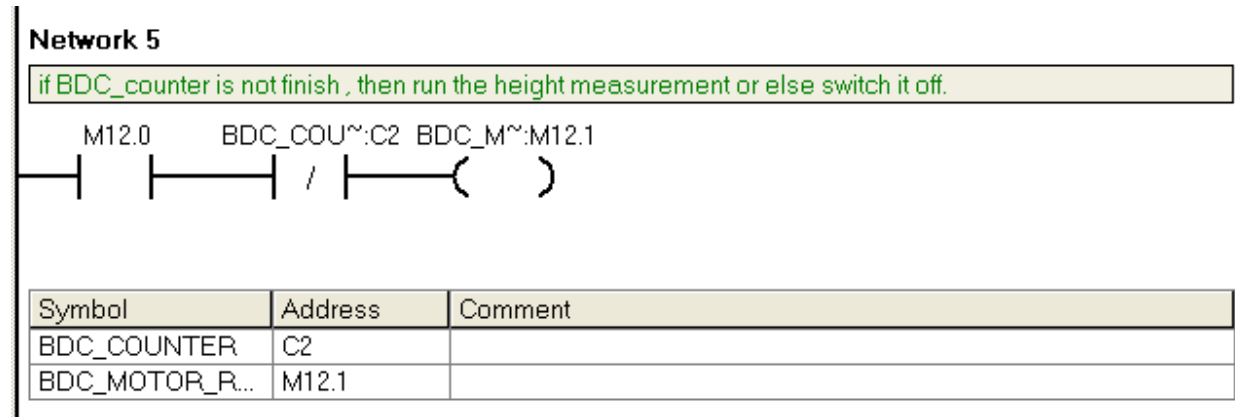
BDC counter used to return the height gauge to the lower dead centre.



Symbol	Address	Comment
BDC_COUNTER	C2	

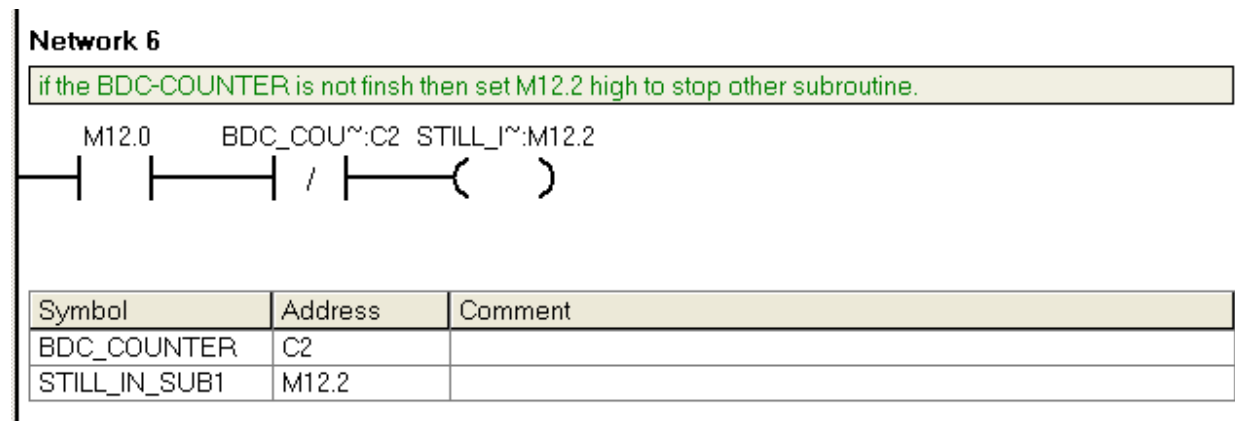
Network 5

This network is used to run the height motor until the height gauge has reached the BDC.



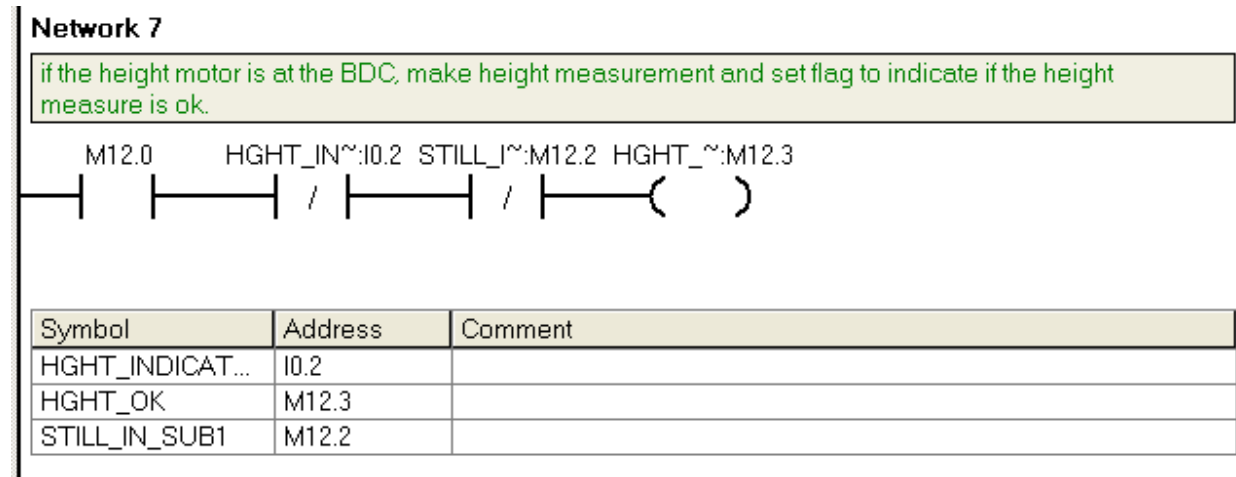
Network 6

In this network, the Flag M12.2 (STILL_IN_SUB1) is set, thus blocking the operation of other subroutines until it has finished (reaches the BDC).



Network 7

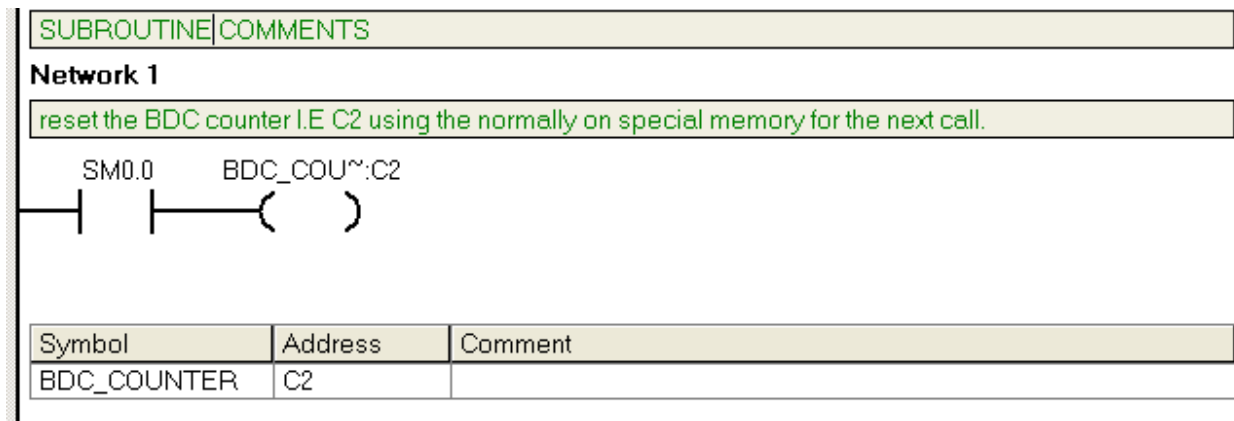
As discussed in the main program, if the height motor has successfully being placed at the BDC and that the height of the washer is within the specified ranges of the height. The height of the washer is found to be ok (achieved by HGHT_INDICATOR) hence, set the HGHT_OK FLAG (M12.3) otherwise, there is no need to set any flags if the height of the washer is not within the limit (Not gauge ok).



5.3.4 Subroutine 2 (SBR_2)

Network 1

This network resets the BDC_COUNTER (C2) that was used to bring the height gauge motor to the Button Dead Centre and thus there will be no more indication that the height motorised gauge is still at BDC to allow the subroutine 1 to be called again (return the height motor to TDC).

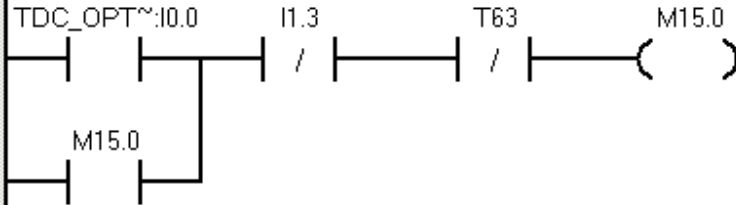


Network 2

Switch the Total Dead Centre (TDC_OPTO) on to capture and the latch the memory M15.0 to be timed out by T63.

Network 2

switch the Total Dead Centre (TDC) opto on to capture and the latch the to the memory M15.0 to be timed out by T63.



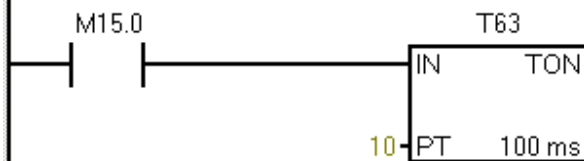
Symbol	Address	Comment
TDC_OPTO	I0.0	

Network 3

T63 is used to reset counter 3 (TDC_COUNTER_2) in about 2 seconds. This should operate, provided that the height motor is at BDC.

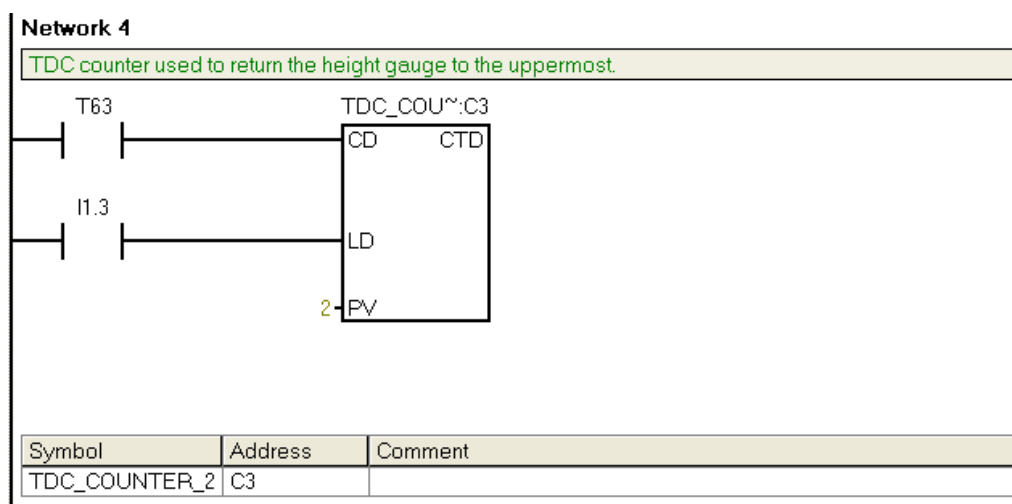
Network 3

T63 is used to count down in about 2 seconds.



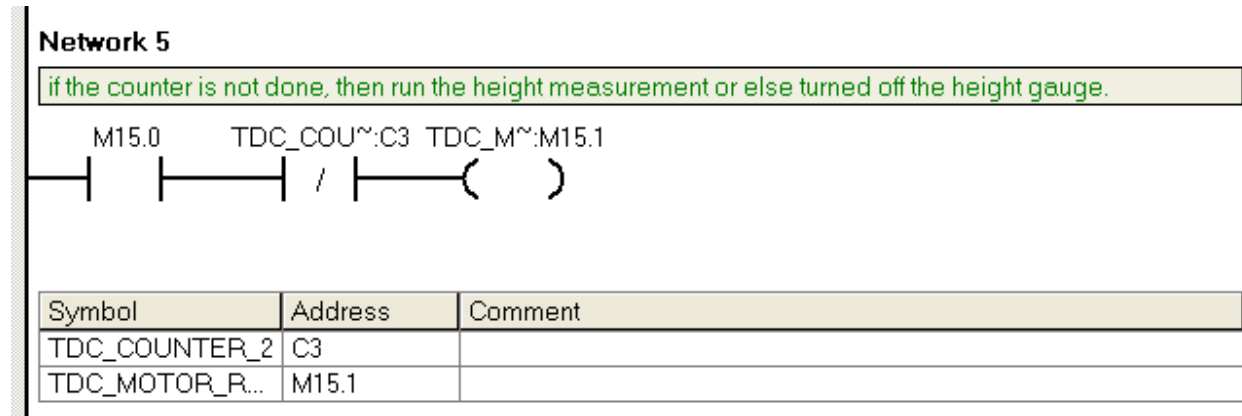
Network 4

This network operates in conjunction with network 3 to return the height gauge to the uppermost.



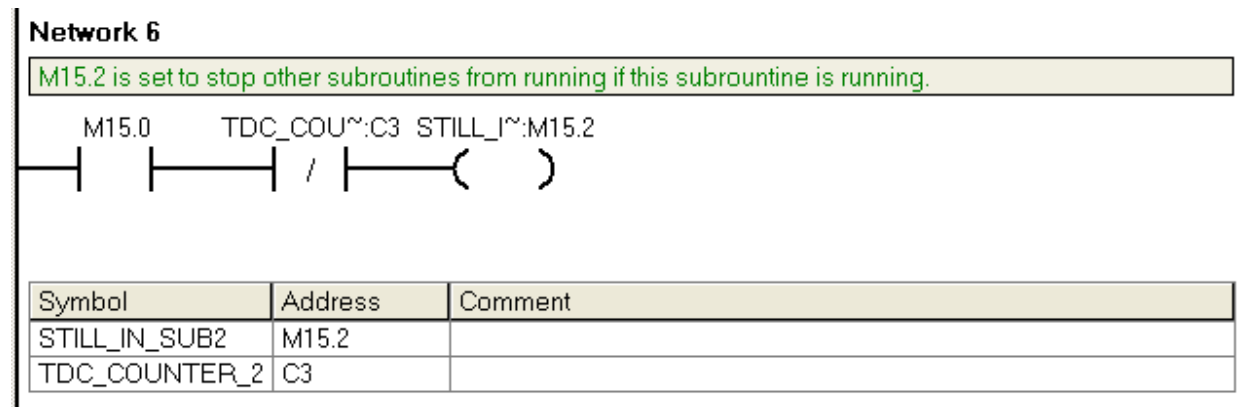
Network 5

This network is used to run the height measurement or else the height gauge is turned off..



Network 6

As discussed in the main program, each of these subroutines can only run one at a time and cannot run all at the same time. Thus, flag M15.2 (STILL_IN_SUB2) is set to block other subroutines from running until the counter 3 has being reset (TDC_COUNTER_2).



5.4 Test procedures

5.4.1 INPUTS

TDC	I0.0
BDC	I0.1
HEIGHT INDICATOR	I0.2
START	I0.5
STOP	I0.6
OPTO1	I0.3
OPTO2	I0.4

5.4.2 OUTPUTS

DISPENSE SOLENOID	Q0.3
HEIGHT MEASUREMENT MOTOR	Q0.2
CONVEYOR	Q0.4
PLASTIC BIN	Q0.5
METAL BIN	Q0.6

The testing procedures come as result of the mechanical failure of the stop and start button. Hardware simulation is therefore the best option to demonstrate the operation of conveyor subsystem. This is achieved using cases in other words, the possible scenarios which can be

found in the appendix C. However, below is a brief scenario to achieve the operation of the system.

5.4.3 Subroutines

The main program will not be able to run without the initialisation of the subroutines. These subroutines contribute to the whole system greatly. That is, initialising the height motor gauge Subroutine 0 (SBR_0).

This subroutine is only run at the start of the conveyor. The TDC_OPTO (I0.0) is switched on thus, set the TDC flag (M20.0). The TDC counter is then reset using switch (I1.3). The counter can be reset either by I, Q, M or even timers however, the choice of I was made to make it much easier.

Subroutine 1 (SBR_1)

This subroutine lowers the height motor gauge to its BDC. The BDC input (I0.1) is switched on to set the BDC flag (M12.0). The BDC counter is also set by (I1.2). Nevertheless, the range of the height can be checked and test simply by switching on the height indicator input (I0.2) so that the height can be set.

Subroutine 2 (SBR_1)

This subroutine runs after the subroutine 1 to return the height motor gauge to its uppermost. The TDC_OPTO(I0.0) enable, hence, setting the TDC flag. I1.1 is used to reset TDC counter.

5.4.4 Main program

Starting and stopping the conveyor belt:

Starting the conveyor is achieved by pushing the closed switch (I0.5) to latch the start button (START_PRESS) and set flag (M30.0). This Flag M30.0 will be able to latch the START_FLAG which will be used more often through the whole system. The system can be manually stopped (halted) simply by pushing the normally closed button (I0.6) and thus, clear the START Flag to stop the conveyor subsystem.

5.4.5 First optical sensor

The detection of incoming washers alerts the height motorised gauge is achieved by switching the physical input (I0.3) which is an OPTO_1 input. This will set RUN_ON_LATCH flag in conjunction with T100 AND T36 timers to stop the conveyor at the right position, at the right time and allow enough time to perform height measurement.

5.4.6 Second Sensor

The second sensor (OPTO_2) is used to detect the material type as either metal or plastic. This process is achieved by manually switching on the physical input (I0.4). This will set optical 2 flag (AT_OPTO_2) that will be used during the detection process. However, some considerations are needed in this procedure. Firstly, if the height of the washer gauge is to be not ok then there is no need to do more detection and the washer will be directed to the dumping bin. The second and third scenarios are that if their height is within the predefined range, then the detection of material type is needed. As mentioned previously in the subroutine (SBR_1), if the height is ok, then set the HT_OK_AT_OPTO_2 flag. The combination of AT_OPTO and HT_OK_OPTO_2 will set the HGHT_OK flag. The normally closed input is also used as the METAL_PLASTIC and should be active if this input is physically pushed. The detected metal will be directed to its metal bin (Q0.6) using the period determined by T33 timer as well, the detected plastic will be directed to its plastic bin (Q0.5) during the period determined by T97 time.

5.4.7 Manual Demonstration Procedures for Conveyor Belt Subsystem

Subroutines

Switch on TDC input (I0.0)

Switch on BDC input (I0.1)

Make sure HGHT_INDICATOR is off

Main program

Make sure all switches are on

I.E

Switch on OPTO_1 (I0.3)

Switch on OPTO_2 (I0.4)

Start the system by pushing START_PRESS

System can be halt be pushing STOP_PRESS

Chapter 6

6.0 SCADA system for the conveyor

6.1 Introduction

The aim of this project involved the implementation of PLC for the conveyor belt system as well as the controlling of the conveyor belt system using the Citect/SCADA software. This software has mechanism that uses HMI to control the PLC. This section was not completed in this project but this chapter will outline some the steps that could have being done if time would have permitted. Based on the given examples and some of the previous studies made by some students that it was a clear understanding of the expectations that would have being proven if there was enough time available. For consideration, the process of uploading the program needs PLC to be on the run mode and the SCADA system provides a toggle click button on the computer screen to run/stop the conveyor system outputs. The basic step that would have being done to achieved this section is the setup of the communication between PLC and SCADA with emphasis on the express communication wizard, communication dialogs, the testing of communications.

6.2 The setup of communication between PLC and SCADA

To implement the controlling procedures, the MIMIC diagram has to be created or built on the Citect Graphic Builder but it is very essential to run this Program on SCADA. Nevertheless, this program cannot run if there is no communication between Citect/SCADA and PLC and thus, there is need to setup the communications between the PLC and Citect/SCADA.

6.2.1 Express Communications Wizard

The purpose of the Express communications wizard is that it allows the quick setup

communication with new or existing I/O device. For every Citect project there is need of at least one I/O server that is defined for it to communicate the I/O devices. Every I/O device should be setup in Citect within appropriate communications protocol and settings and any interface boards and communication ports in the computer has to be defined as well. The following bullet points, step through the Express Communications Wizard:

- I/O server names
- Interface Boards in the computer
- Communications Ports on the boards
- I/O devices attached to the port

Running the Express Communications Wizard:

In order to run the Express Communications Wizard, go to Citect Explorer, and then select the project then double- click on the communication folder and double-click on the Express I/O device Setup. For clarification, the source of below figures is Citect SCADA Help manual.

1. In the Citect Explorer select the name conveyor project.
2. Double click on the communications folder and double click on Express I/O device Setup.
This will allow the below dialog to appear.

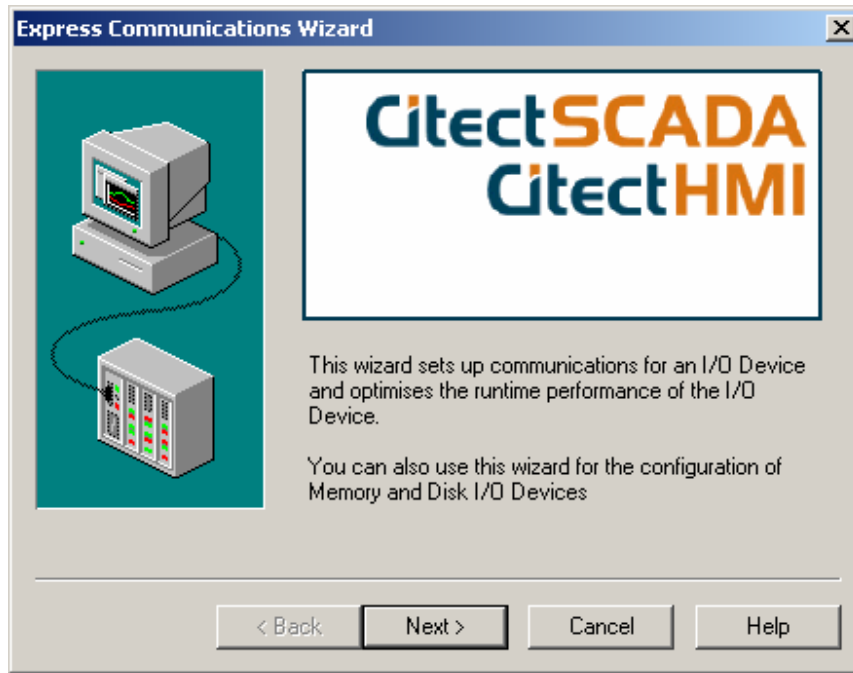


Figure 6.2 (a) Express Communication Wizard

3. Click next to continue and the below dialog box should appear

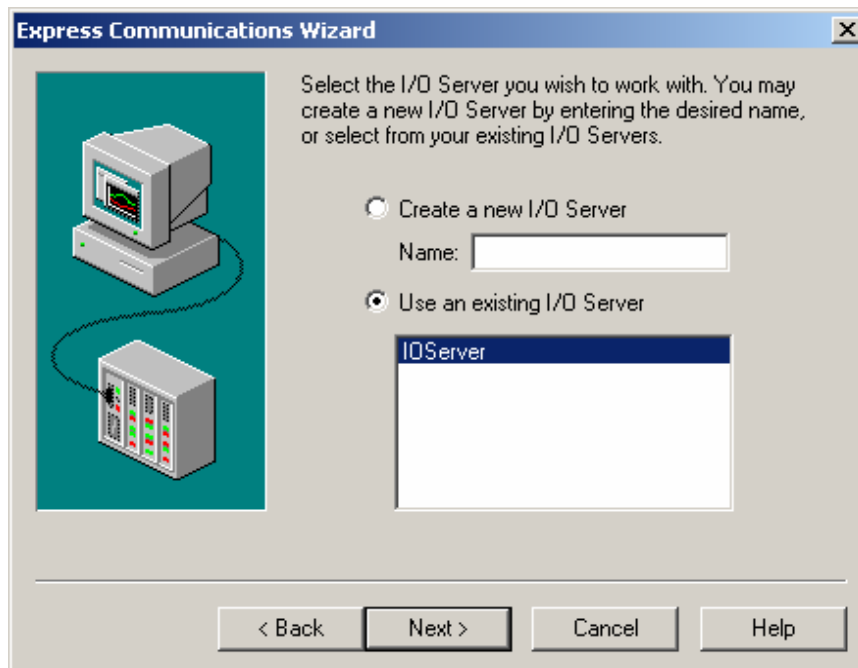


Figure 6.2 (b) Express Communication Wizard with I/O Server Name

4. Leave the above default name as “IOServer” and click next to continue and the next dialog box will appear.

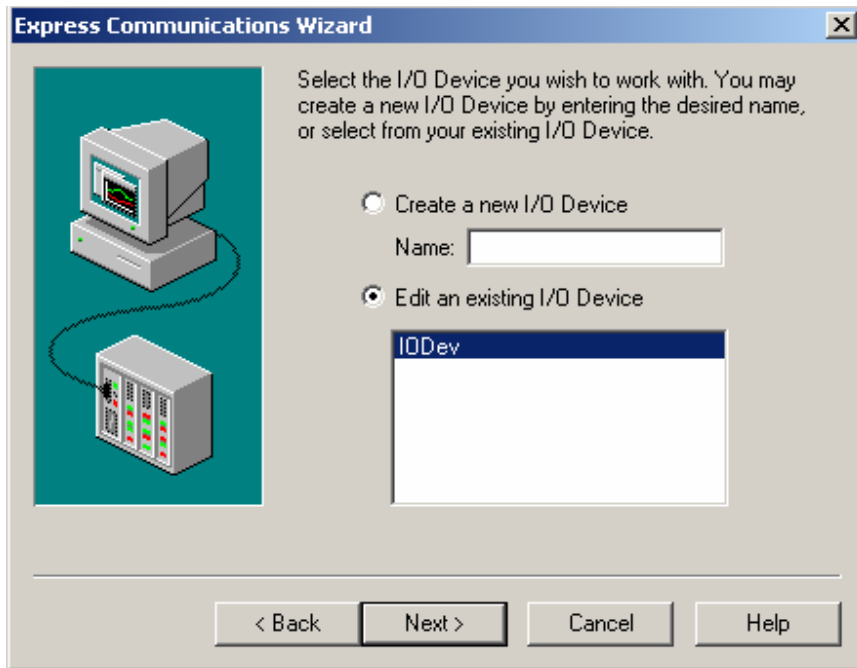


Figure 6.2 (c) Express Communication Wizard with Edit I/O Device Option

5. An existing I/O Device name will appear then click next to continue. The Following dialog will appear.



Figure 6.2 (d) Express Communication Wizard with Different I/O Device

6. Select External I/O Device. Then select the Manufacturer, Model and Communications Method from the list. Click next to continue

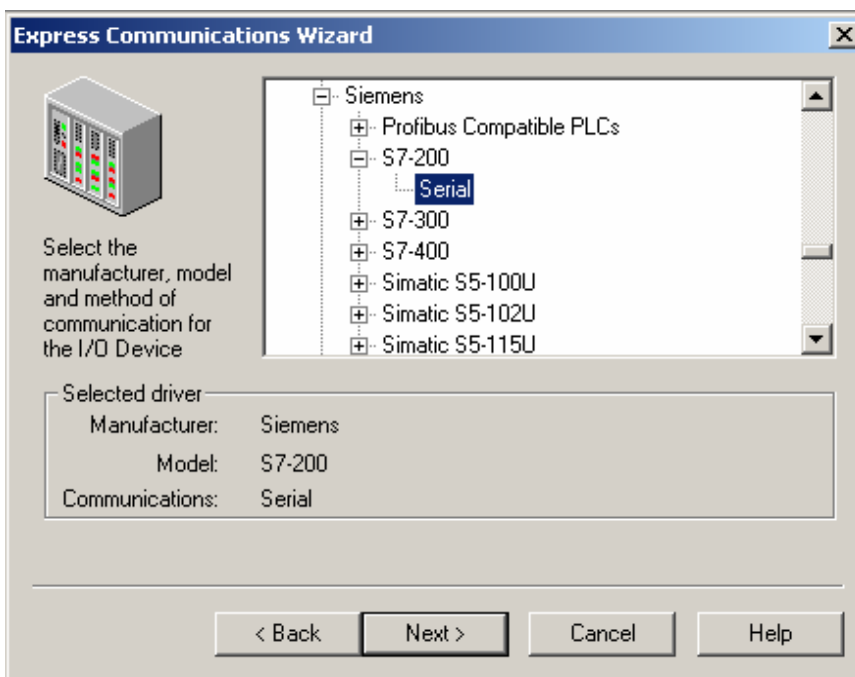


Figure 6.2 (e) Manufacture, Model and Method of Communication

7. The next dialog will appear if connected to a serial device and allows setup a modem connection to I/O Device. Click next to continue.

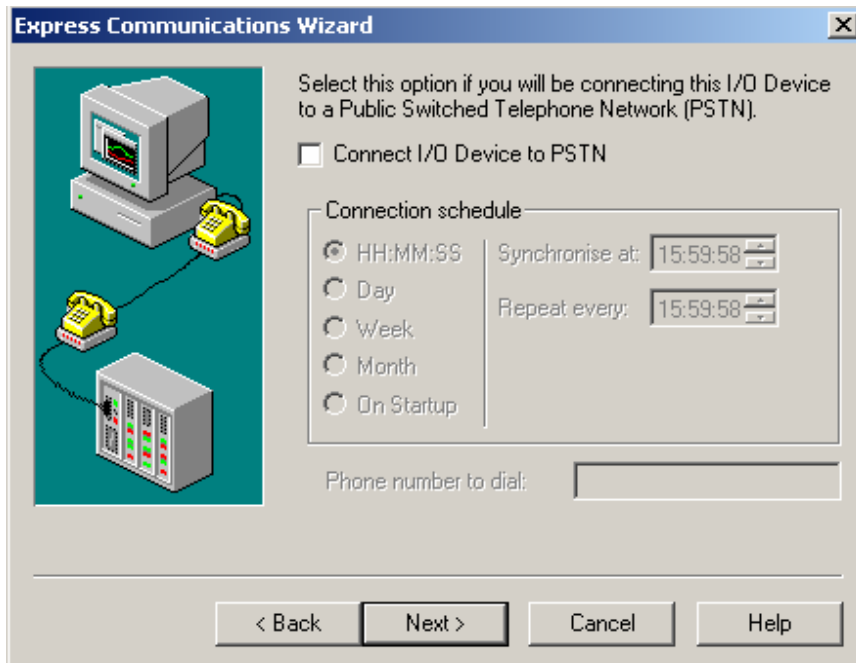


Figure 6.2 (f) Public Switched Telephone Network Option

8. This dialog allows entering the com port for serial device. Select the correct com port. In this project the only detected serial port will appear (COM1). Click next to continue.

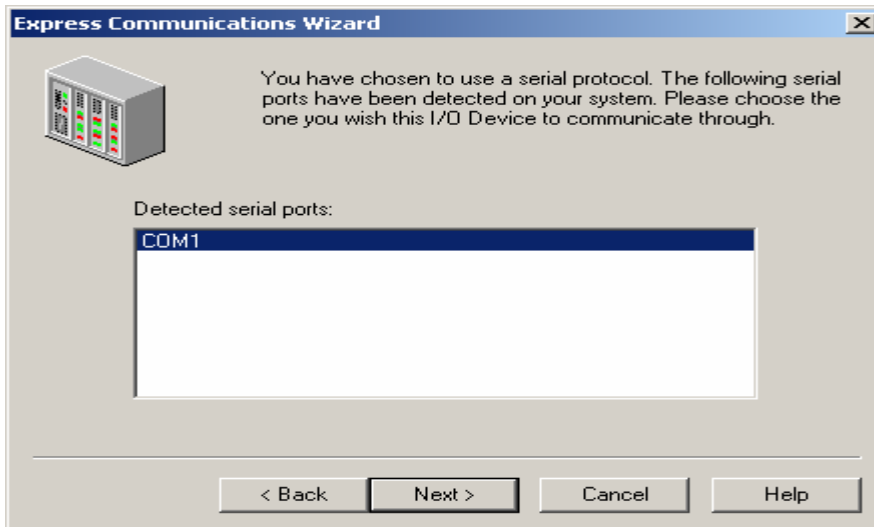


Figure 6.2 (g) Serial Ports Detected on System

9. The next dialog box allows to link to an external database containing variable tag information. The link can be to a static file or it may take advantage of Microsoft's COM and DCOM technologies. In this project there is no need to use this feature either. Click next to continue

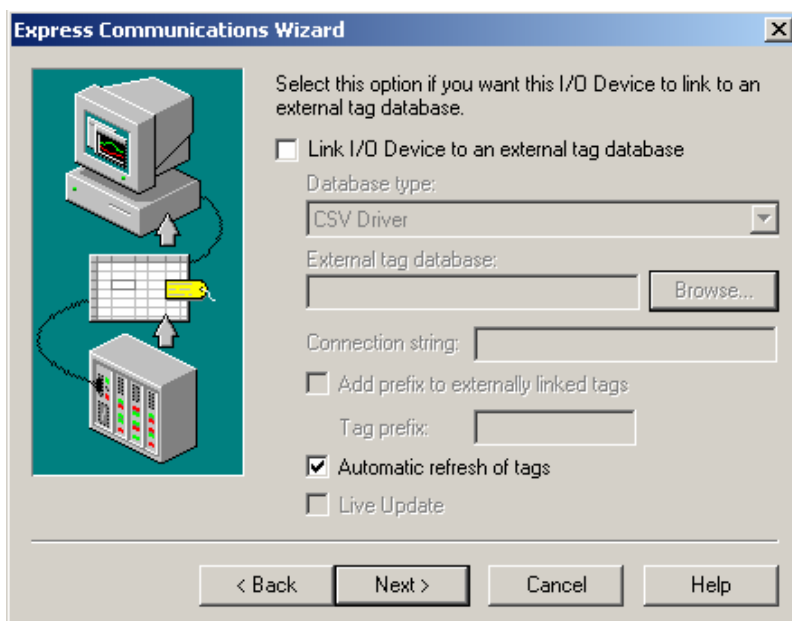


Figure 6.2 (h) External Tag Database Option

10. The final dialog displays a summary of the communications setup. Click finish to end up express communications wizard.



Figure 6.2 (i) Final Step for Express Communication Wizard

6.2.2 Communications Dialogs

After running the Express Communications Wizard, the communications dialogs should have been created as well for the project. These are opened from the Citect Explorer which is in the communication folder or it should be in Citect Project Editor achieved by selecting the menu communication. This should display the dialogs of I/O server, Ports, Boards, and I/O devices.

6.2.3 Testing Communications

It is much recommended that the project should be tested in the early stage during the development of the project. This contributes to the fact that the project will not operate to satisfy

the expectations if there is no reliable communications. This test of communications also ensures that the correct addressing format was used for the variable Tags that were defined in the project.

1. Define a variable tag called Test.

- a. In Citect Explorer, Select the **S7-200 Communications** project and open the Tags folder then double click on Variable Tags. The following form will be displayed.

The screenshot shows a dialog box titled "Variable Tags [S7-200 communication test]". It has several input fields and dropdown menus. The "Variable Tag Name" field contains "TEST". The "Data Type" dropdown is set to "DIGITAL". The "I/O Device Name" dropdown is set to "IODEv". The "Address" field contains "MR1.0". There are also empty fields for "Raw Zero Scale", "Raw Full Scale", "Eng Zero Scale", "Eng Full Scale", "Eng Units", and "Format". At the bottom, there are four buttons: "Add", "Replace", "Delete", and "Help". The status bar at the bottom left shows "Record: 1" and at the bottom right shows "Linked: No".

Figure 6.2 (j) Variable Tag for Communication Test

- b. Fill in the fields of the form as follows:

Variable Tag Name	Test
Data Type	DIGITAL
I/O Device Name	IODEv
Address	MR1.0

- c. Click on **Add**.

2. Create a new graphics page to display the value of the Test tag.

- a. In Citect Explorer, select the S7-200 Communications project and open the Graphics folder then open the Pages folder.

- b. Double click on the Create a new page button.

- c. To draw a button on the page, select the Button tool from the toolbox.
- d. Move the mouse to where you want the button to start and press the left mouse button
Drag the mouse to where you want the button to finish and release the mouse button.
- e. The following dialog will displayed.

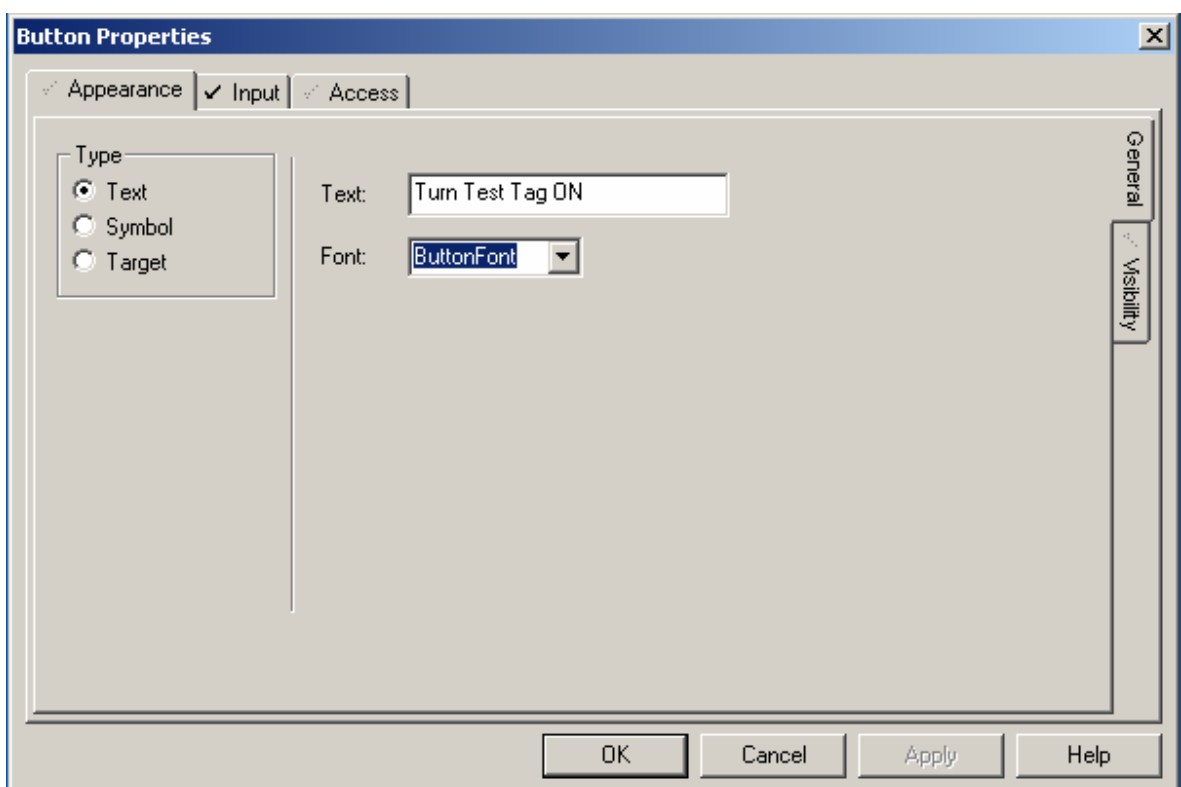


Figure 6.2 (k) Button Properties for Communication Test

- f. Click on the Input tab (at the top of the dialog) and the following dialog will be displayed.

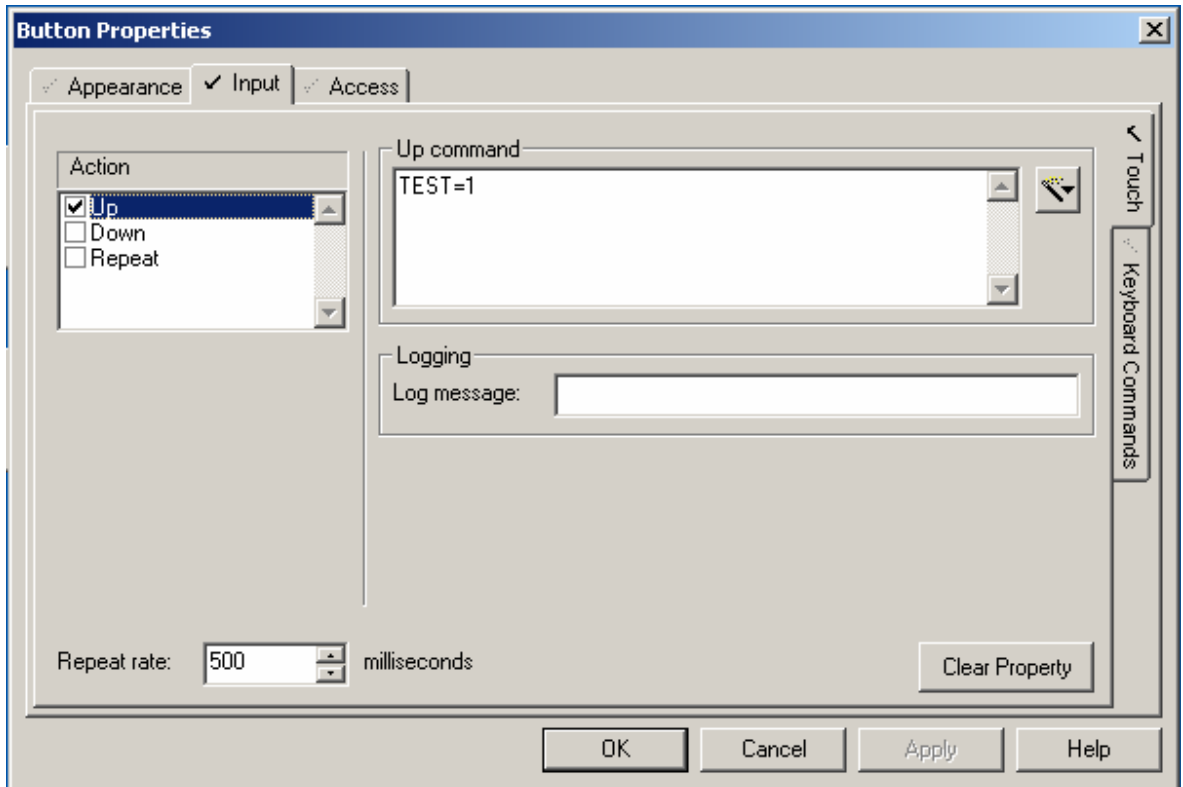


Figure 6.2 (l) Button Properties Up Command for Communication Test

In the Action list click the Up check box and in the Up command field
Type **Test = 1**.

g. Click OK.

h. Draw another button by repeating steps e to h. except for this button, type Turning Test
Tag Off in the Text field and Test=0 in the Up command field.

i. Select the Text tool from the toolbox and move the cursor to a point near the
buttons you drew – by doing this we will place some text on the page to display
the value of the Test tag.

j. Press the space bar once and then click the left mouse button. A dialog will be
displayed.

k. Click on the Appearance tab along the top of the dialog then click on the display Value tab at the right of the dialog. The following dialog will be displayed.

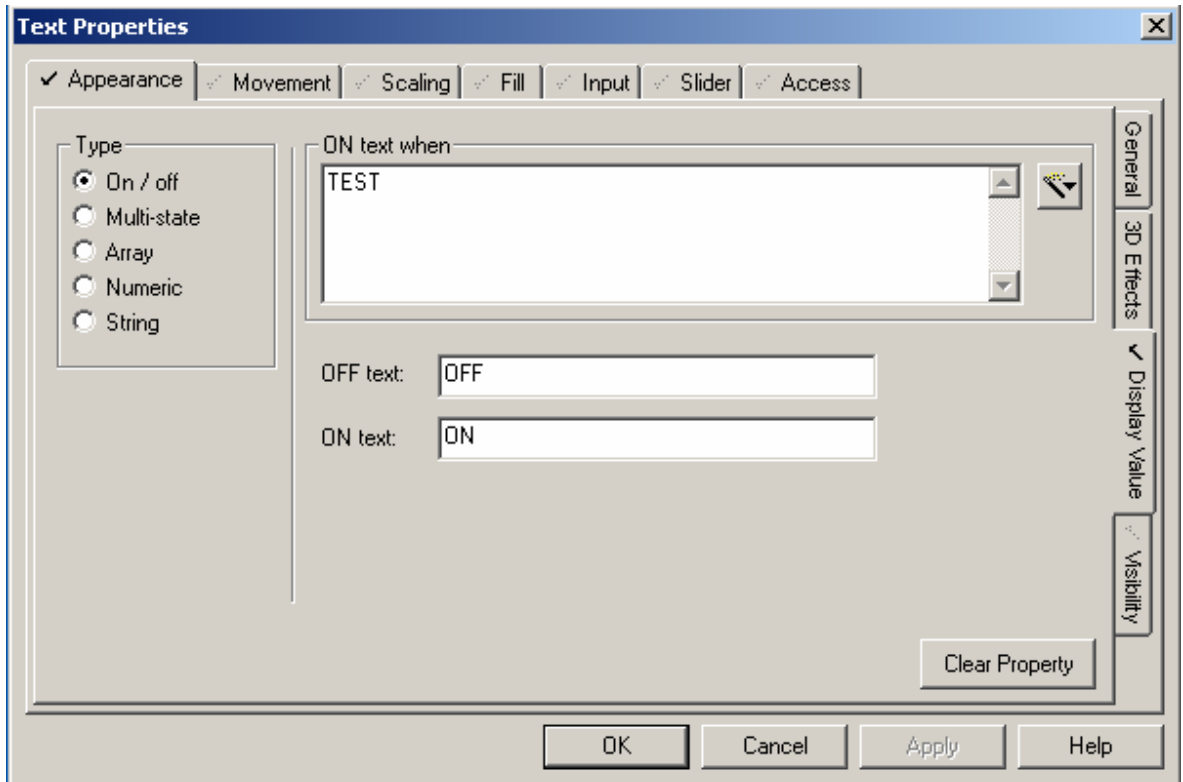


Figure 6.2 (m) Text Properties for Communication Test

In the ON text when field type Test (as a default the Type should already be set as On / Off), in the OFF field type OFF and in the ON text field type ON.

L. Click on OK. Your page should now look similar to the following. Then click on the Save button on the tool bar to save the page.

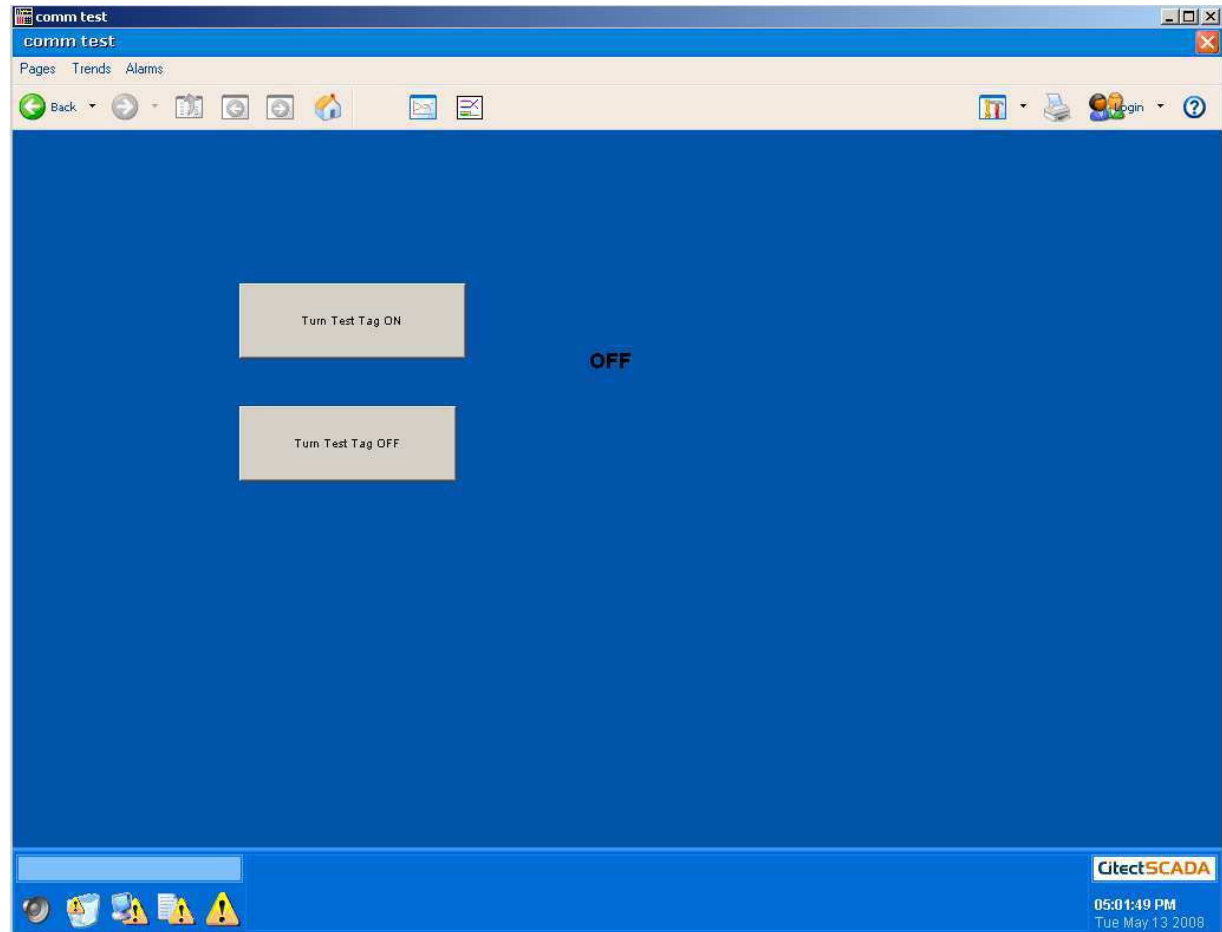


Figure 6.2 (n) MIMIC Diagram for Communication Test

3. Run the Computer Setup Wizard.

- a. In Citect Explorer, click on My Projects in the Project List and then double click on Computer Setup in the Contents Area.
- b. Choose Express Setup mode.
- c. Select Stand-alone Computer (server and display client).
- d. Select the S7-200 Communications project from the drop down list.

e. As there is only one I/O Server defined in the S7-200 Communications project, it will automatically be selected. Click next.

f. Click Finish to complete the setup and save changes.

4. Compile and run the project and check you have communications.

a. In the Project Editor, click on the Run Project button. This first compiles then runs the project.

b. The project will now continue to run and accept mouse and keyboard input until it is shutdown.

d. Check that you have communications and can change the value of the Test tag. If the I/O Server and I/O Device are not communicating, then **#COM** will appear where the text was drawn.

e. To shutdown the project, click on the close box on the title bar of the project window.

Chapter 7

7.0 Discussion and Conclusion

This chapter as mentioned previously will discuss the achievements of the objectives, which will provide the details of the achieved objectives of this project as well as the remaining work that needs further study. The conclusion will summarise the basics of this project.

7.1 Achievements of objectives

This section details the objectives of this project which as being achieved. As provided in appendix A, the following project objectives were stated.

- To become familiar and confident with basic concepts of ladder logic.
- To convert the existing Allen Bradley Micrologix 1000 PLC based controller into Siemens S7200 PLC based Controller.
- To familiarise with the Citect supervisory control and data acquisition software(SCADA).
- To construct a mimic diagram using Citect to allow operation of the conveyor from a PC.

As per first objective, familiarity and confidentiality of understanding of the basic ladder logic was achieved. The basic of ladder logic were successfully understood. These included the basics such as inputs, outputs, counters, timers, interrupts and subroutines. The details of these basics of the Siemens PLC based control are provided in chapter 3 which provides the details of PLC hardware and Software. The

second objective was the conversion of the existing Allen Bradley Micrologix program into the new standardized PLC known as Siemens S7200 based control. The new program is written using the new PLC software and was tested to be successful. The achievements of this objective include simulation of the written program using hardware simulation and detailing each network as well as providing the function of each network. The third objective deals with understanding the basics of SCADA. This objective was accomplished as it can be proven by referring to chapter 4 which provides the details of SCADA, its features are used to achieve the control of the system as well the simple examples provided to demonstrate the understanding of this objective.

7.2 Further work

This project deals with industrialised software and thus, there is expectation that some parts cannot be achieved due to either time constraints or more study is required. The last of the objectives for this project was not achieved due to the time limit. The aim of this objective was to construct the mimic diagram in which conveyor could be control and operates through PC. This operation of this part was attempted but it could not be finished to meet the dateline.

7.3 Conclusion

This dissertation covered the backgrounds of both special industrialised softwares' (PLC and SCADA) that are used in power engineering, process engineering, power generation and some many industries that deals with the control systems. The overview of the conveyor system was also covered in which the background of electromagnetic actuators, optical sensors were provided as well the emphases on the eddy current which enables the detection on the type of material. This project is very essential for the education purpose as it provides the knowledge with respect to many disciplines in engineering.

References

Alan J. Crispin, 1990, *Programmable Logic Controllers and their Engineering Applications*- McGraw-Hill International (UK) Ltd.

Australia.rs-online.com (manufacturing website for R-S actuators)

Citect 2001, Citect configuration Training Manual, NSW, Australia

David Bailey & Edwin Wright, 2003, *Practical SCADA for Industry-IDC Technologies*.

HMI Citect SCADA, Version 6.0, Citect SCADA Help

John W. Webb & Ronald A. Reis, 1999, *Programmable Logic Controllers – Principles and Applications*, 4th edition – Prentice Hall, IDC

Jon Stenerson, 1999, *Fundamentals of Programmable Logic Controllers, Sensors and communications*, 2nd edition- Prentice Hall, IDC

L. A. Bryan & E.A. Bryan, 1997, *Programmable Controllers- Theory and applications*, 2nd edition – Industrial Text Company

Peter Oxenham, 2001, *PLC Based PID Control Supervised by SCADA*

Sabrie Soloman, 1998, *Sensors Handbook*

Scott D. Lathrop, Christopher L. Gates, Darrell D. Massie, John M.D. Hill - *Risk Assessment of a Power Plant: Evaluating the Security of a Supervisory Control and Data Acquisition System*

S. Middelhoek, J. Van Der Spiegel, 1987, *Sensors and Actuators*

Siemens Schematic, S7200, Programmable Controller System Manual

T. Obina, C.O. Pak, Y. Sato, A. Mishina, K. Harada, Y. Kobayashi, T. Miyajima, S. Nagahashi, T. Nogami, S. Sakanaka, T. Shioya, M. Tadano, T. Takahashi, Y. Tanimoto, and K. *Umemori Upgrade of the Photon Factory Control System*

W. Bolton, 2006, *Programmable Logic Controllers*, 4th edition- Elsevier Ltd

APPENDIX A- PROJECT SPECIFICATIONS

ENG4111/4112

PROJECT SPECIFICATION

FOR: Reec R. Muorwel

TOPIC: Remotely Controlled Material Handling Conveyor Belt

SUPERVISOR: Dr. Tony Ahfock

PROJECT AIM: To design and test a SCADA and PLC based remotely controlled material handling conveyor belt.

PROGRAMME: (Issue A, 25th March 2008)

- 1) To become familiar and confident with basic concepts of ladder logic

- 2) To convert the existing Allen Bradley Micrologix1000 PLC based conveyor belt controller into a Siemens S7200 PLC based controller.

- 3) To familiarise with the Citect supervisory control and data acquisition software (SCADA).

- 4) To construct a mimic diagram using Citect to allow operation of the conveyor from a PC.

To use the Sun Global Desktop (SGD) software to remotely operate the controlled material handling conveyor belt by means of a Citect SCADA

AGREED _____ (student) _____
(supervisor)

Date / /2008

Date / /2008

Examiner/Co-examiner: _____

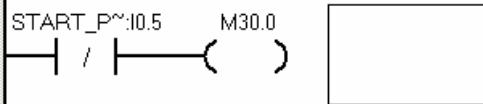
APPENDIX B- PLC LADDER DIAGRAM

Main Program

Main Program

Network 1

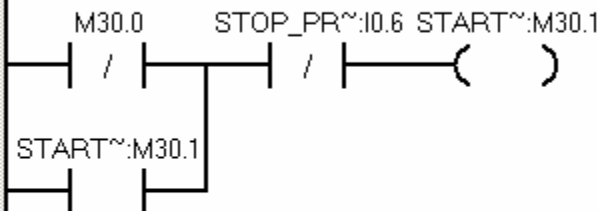
set flag M30.0 to latch start button



Symbol	Address	Comment
START_PRESS	I0.5	

Network 2

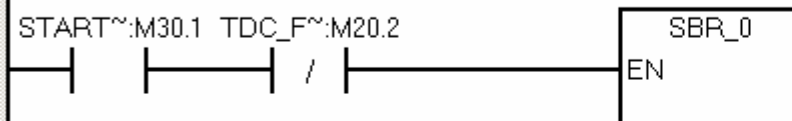
latch the start flag to be running until the stop_press is pushed.



Symbol	Address	Comment
START_FLAG	M30.1	
STOP_PRESS	I0.6	

Network 3

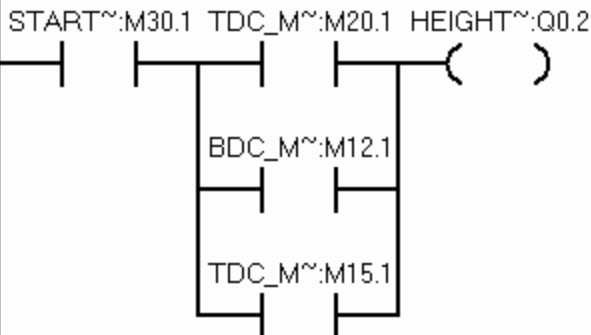
set the height notor gauge to it TDC at start only by calling the subroutine 0 (SBR_0)



Symbol	Address	Comment
START_FLAG	M30.1	
TDC_FLAG	M20.2	

Network 4

M20.1 is set initially as TDC_MOTOR_RUN by subroutine 0 (SBR_0) to be run only at start.
 M12.1 is set initially as BDC_MOTOR_RUN by subroutine 1 (SBR_1) to bring height motor to BDC
 M15.1 is set initially as TDC_MOTOR to return height motor to it uppermost.
 Run height motor if one of the above subroutines is running.



Network 5

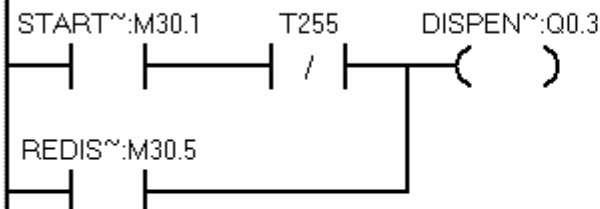
dispense one washer on the conveyor at the start I.E turn on the dispense soleniod at the start press



Symbol	Address	Comment
START_FLAG	M30.1	

Network 6

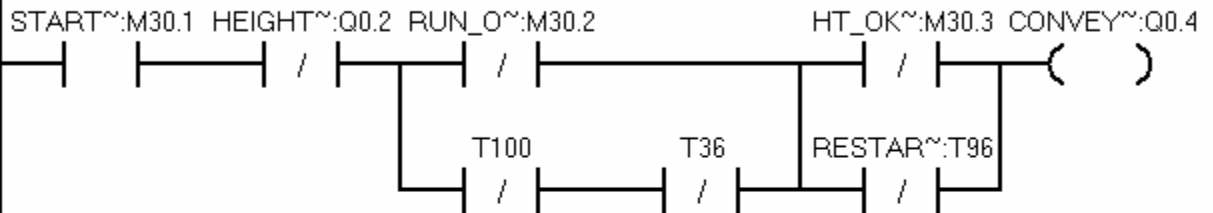
energised the dispense soleniod at start press (on for 2 seconds and de-energised) and progressively after the detection of metal/plastic (OPTO_2) acheived by Redispende



Symbol	Address	Comment
DISPENSER	Q0.3	
REDISPENSER	M30.5	
START_FLAG	M30.1	

Network 7

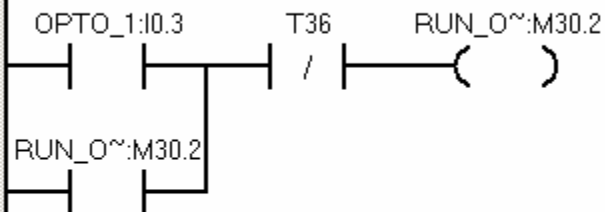
at start, energised the conveyor and stop the conveyor for period of time to make height measurement and if the height is ok . Also conveyor is to be stop at OPTO_2 for plastic and metal detection.



Symbol	Address	Comment
CONVEYOR	Q0.4	
HEIGHT_MOTOR	Q0.2	
HT_OK_AT_OPT...	M30.3	
RESTART_CON...	T96	
RUN_ON_LATCH	M30.2	
START_FLAG	M30.1	

Network 8

first optical (OPTO_1) is latch to detect the incoming washer



Symbol	Address	Comment
OPTO_1	I0.3	
RUN_ON_LATCH	M30.2	

Network 9

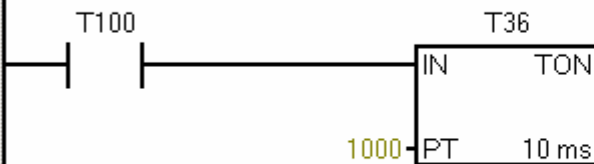
if the incoming washer has passed the OPTO_1, T100 is used to place the washer at the right time under the height measurement device to performance height measurement.



Symbol	Address	Comment
RUN_ON_LATCH	M30.2	

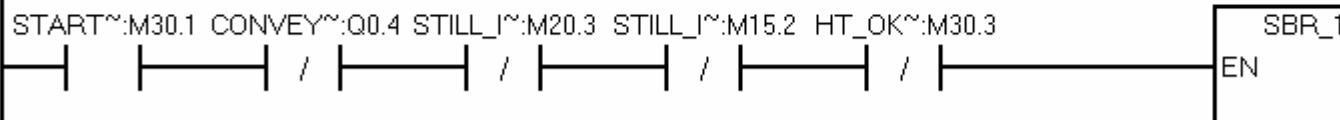
Network 10

T36 is then used to hold the conveyor from running until the height measurement is accomplished.



Network 11

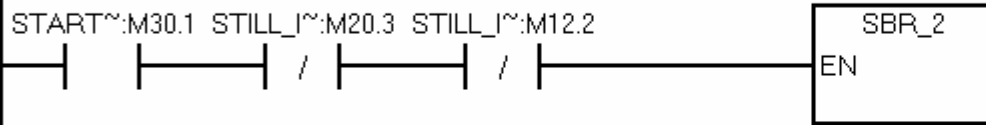
call the subroutine 1 (SBR_1) to bring height motor to BDC(lower the height motor) only if the conveyor is not on, not still in subroutine 0 (SBR_0) as well not running the subroutine 2 (SBR_2)



Symbol	Address	Comment
CONVEYOR	Q0.4	
HT_OK_AT_OPTO...	M30.3	
START_FLAG	M30.1	
STILL_IN_SUB0	M20.3	
STILL_IN_SUB2	M15.2	

Network 12

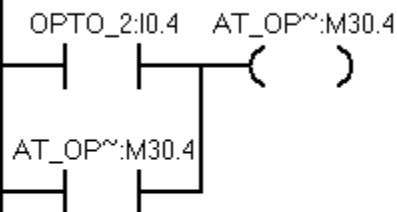
after the completion of height measurement(SBR_1), return the height motor to it uppermost not to interfere with the conveyor. this achieved by call subroutine 2 (SBR_2) only if it is not still in SBR_0 or SBR_1.



Symbol	Address	Comment
START_FLAG	M30.1	
STILL_IN_SUB0	M20.3	
STILL_IN_SUB1	M12.2	

Network 13

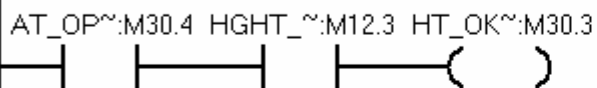
detect the incoming washer at the second optical (OPTO_2) and set it target (AT_OPTO_2)



Symbol	Address	Comment
AT_OPTO_2	M30.4	
OPTO_2	I0.4	

Network 14

if the AT_OPTO_2 is high and the height measurement was ok(within the specified range) then stop the conveyor to perform metal/plastic detection for about a period of two second which is timed out by T96



Symbol	Address	Comment
AT_OPTO_2	M30.4	
HGHT_OK	M12.3	
HT_OK_AT_OPT...	M30.3	

Network 15

restart the conveyor if the detection is done



Symbol	Address	Comment
HT_OK_AT_OPT...	M30.3	
RESTART_CON...	T96	

Network 16

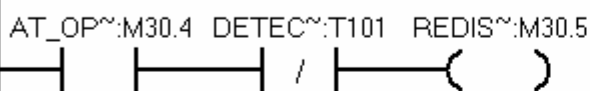
as the incoming washer reaches the OPTO_2, dispense the next washer and is timed out by by T101



Symbol	Address	Comment
AT_OPTO_2	M30.4	
DETECT_ME	T101	

Network 17

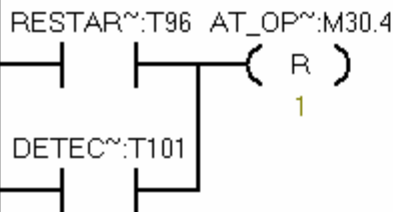
redispense if the T101 is not done



Symbol	Address	Comment
AT_OPTO_2	M30.4	
DETECT_ME	T101	
REDISPENSER	M30.5	

Network 18

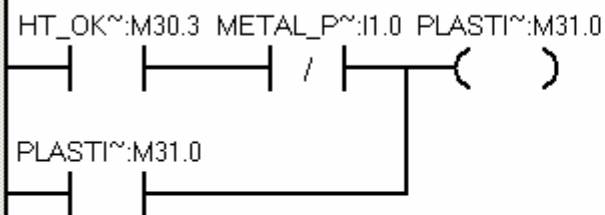
reset the AT_OPTO_2 to clear and able to allow another pass.



Symbol	Address	Comment
AT_OPTO_2	M30.4	
DETECT_ME	T101	
RESTART_CON...	T96	

Network 19

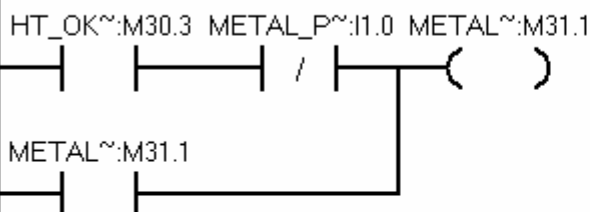
if height measurement was ok (HT_OK) then make metal/plastic detection and set plastic flag if plastic



Symbol	Address	Comment
HT_OK_AT_OPT...	M30.3	
METAL_PLASTIC...	I1.0	
PLASTIC_BAG	M31.0	

Network 20

if the height measurement was ok, take plastic/metal and set metal flag if metal



Symbol	Address	Comment
HT_OK_AT_OPT...	M30.3	
METAL_BAG	M31.1	
METAL_PLASTIC...	I1.0	

Network 21

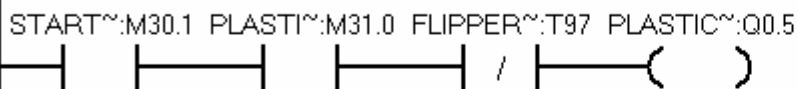
if the height ok and plastic flag has being set, T97 (FLIPPER_2_off) is used for directing the plastic washer to it respected bin



Symbol	Address	Comment
FLIPPER_2_OFF	T97	
PLASTIC_BAG	M31.0	

Network 22

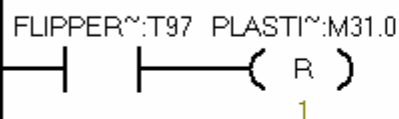
direct plastic washer to it bin



Symbol	Address	Comment
FLIPPER_2_OFF	T97	
PLASTIC_BAG	M31.0	
PLASTIC_BUCKET	Q0.5	
START_FLAG	M30.1	

Network 23

clear the plastic flag if T97 has being timed out to allow the next run.



Symbol	Address	Comment
FLIPPER_2_OFF	T97	
PLASTIC_BAG	M31.0	

Network 24

if the height ok and metal flag has being set, T33 (FLIPPER_1_off) is used for directing the metal washer to it respected bin

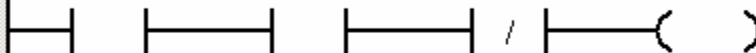


Symbol	Address	Comment
FLIPPER_1_OFF	T33	
METAL_BAG	M31.1	

Network 25

metal washer being directed to it bin

START~:M30.1 METAL~:M31.1 FLIPPER~:T33 FLIPPER~:Q0.6

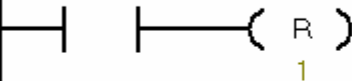


Symbol	Address	Comment
FLIPPER_1	Q0.6	
FLIPPER_1_OFF	T33	
METAL_BAG	M31.1	
START_FLAG	M30.1	

Network 26

clear the plastic flag if T33 has being timed out to allow the next run

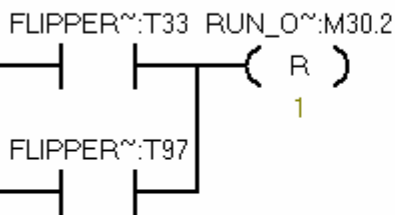
FLIPPER~:T33 METAL~:M31.1



Symbol	Address	Comment
FLIPPER_1_OFF	T33	
METAL_BAG	M31.1	

Network 27

clear all the flags (targets) used at the end of each and every washer to allow clear operation accordingly.



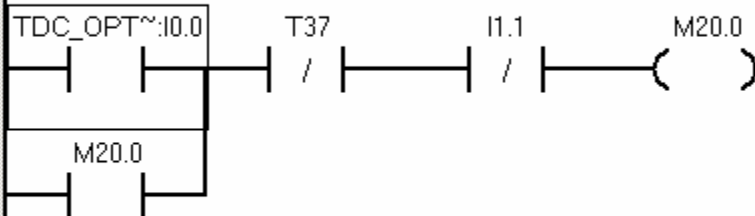
Symbol	Address	Comment
FLIPPER_1_OFF	T33	
FLIPPER_2_OFF	T97	
RUN_ON_LATCH	M30.2	

First subroutine ----- SBR_0

SBR_0

Network 1 Network Title

switch TDC_OPT0 on to capture and latch M20



Symbol	Address	Comment
TDC_OPT0	I0.0	

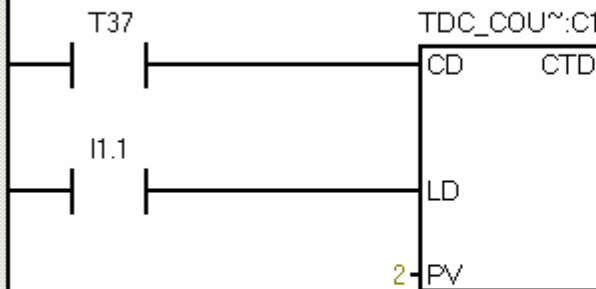
Network 2

T37as input for the counter after every 2 seconds.



Network 3

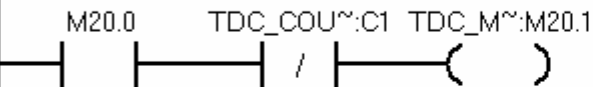
at start, ensure that TDC is at the uppermost for about 2 second.



Symbol	Address	Comment
TDC_COUNTER	C1	

Network 4

run the TDC_MOTOR height motorised gauget if TDC counter is not finish otherwise, switch it off. set M20.1 target which is run only at the start.



Symbol	Address	Comment
TDC_COUNTER	C1	
TDC_MOTOR_R...	M20.1	

Network 5

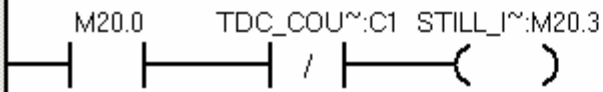
when it reaches the uppermost, set the M20.2 target to stop this subroutine from running unless the the stop button the start have being press.



Symbol	Address	Comment
TDC_COUNTER	C1	
TDC_FLAG	M20.2	

Network 6

set target M20.3 to stop other subroutines to be call if this subroutine is still running.

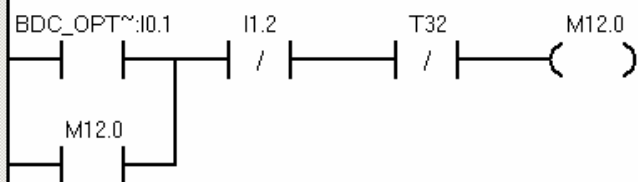


Symbol	Address	Comment
STILL_IN_SUB0	M20.3	
TDC_COUNTER	C1	

Second subroutine (SBR_1)

Network 2

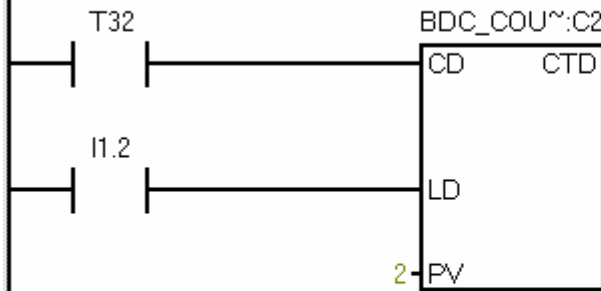
capture and latch if BDC_OPTO is switched on.



Symbol	Address	Comment
BDC_OPTO	I0.1	

Network 4

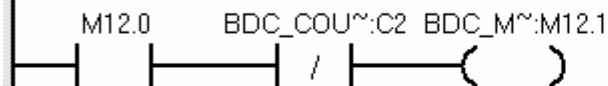
BDC counter used to return the height gauge to the lower dead centre.



Symbol	Address	Comment
BDC_COUNTER	C2	

Network 5

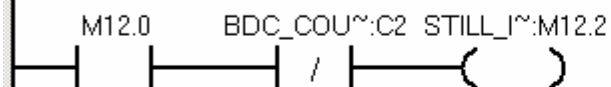
if BDC_counter is not finish , then run the height measurement or else switch it off.



Symbol	Address	Comment
BDC_COUNTER	C2	
BDC_MOTOR_R...	M12.1	

Network 6

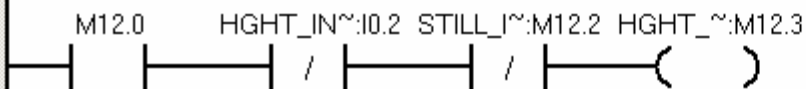
if the BDC-COUNTER is not finish then set M12.2 high to stop other subroutine.



Symbol	Address	Comment
BDC_COUNTER	C2	
STILL_IN_SUB1	M12.2	

Network 7

if the height motor is at the BDC, make height measurement and set flag to indicate if the height measure is ok.

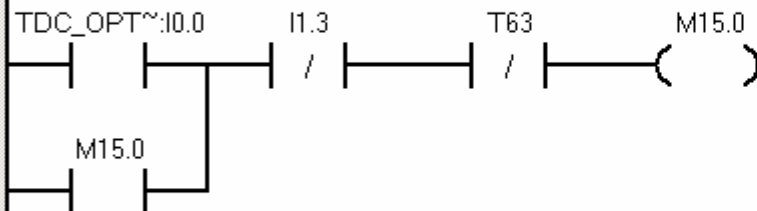


Symbol	Address	Comment
HGHT_INDICAT...	I0.2	
HGHT_OK	M12.3	
STILL_IN_SUB1	M12.2	

Third subroutine (SBR_2)

Network 2

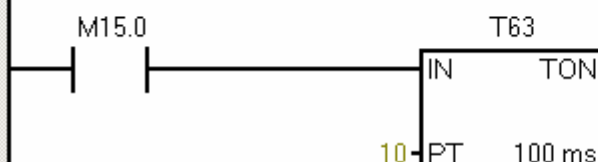
switch the Total Dead Centre (TDC) opto on to capture and the latch the to the memory M15.0 to be timed out by T63.



Symbol	Address	Comment
TDC_OPT0	I0.0	

Network 3

T63 is used to count down in about 2 seconds.



APPENDIX C- DEMONSTRATION SCENARIOS

First Scenario (case 1)

Assume the washer is metal and the height is within the defined height limits

Sequence Number	Operator Action	Expected Outcomes	Comments
1	Switch on I0.0 (TDC_OPTO). Simulation of the first subroutine	This subroutine will be responsible of initial checking either the height motorised gauge is at the TDC or BDC. Pressing I0.0 will latch and set M20.0 to be high thus setting flag TDC_MOTOR_RUN (M20.1). TDC_MOTOR_RUN Flag is set to run the motor only at the start and should not run again until the main program has being halted. TDC_FLAG (M20.2) target is also set to take the height measurement gauge to it uppermost provide that the hight motorised gauge was found to be at the BDC. Finally STILL_IN_SUB0 (M20.3) flag is set to stop other subroutines from running until it is finish.	Refer SBR_0 first subroutine.

2	Switch on I0.1 (BDC_OPTO) simulation of the second subroutine	<p>The expected outcome for this subroutine is to lower the height motorised gauge to it BDC. Normally On special memory (SM0.0) is used to Reset C1 (TDC_COUNTER) so that this counter can be used for the next incoming washer. Pressing I0.1 will latch M12.0 to be ON (HIGH) until it target (flag) is clear. This is achieved by setting the following flags I.E BDC_MOTOR_RUN flag which lower the height motorised gauge to it lowest part and STILL_IN_SUB1 flag which stop other subroutines from running until this subroutine is finished. Most importantly, Height Ok flag (HGHT_OK) is also set to indicate if the height of the washer is within the predefined range of limits. If the HGHT_OK is high, then the detection of the material type will be determined. This will be discussed further in sequence number 6. if the HGHT_OK is not ok,</p>	Refer to SBR_1 second subroutine
---	---	---	----------------------------------

		then there is not need to do any more detection and hence, the washer will be directed to the dumping bin,	
3	Switch on I0.0 (TDC_OPTO). Simulation of the third subroutine	This subroutine is responsible of returning the height motorised gauge to it uppermost (TDC). SM0.0 special memory is used to reset C3 so that it can be recall for the next cycle of incoming washer. To successfully return the Height motorised gauge to it TDC, TDC_MOTOR_RUN flag is set as well setting another flag (STILL_IN_SUB_2) to avoid other subroutines from interrupting while this subroutine is running until finished.	Refer to third subroutine (SBR_2)
4	Press I0.5 (START_PRESS), a normal starting of the conveyor	This will energise height motorised gauge (Q0.2) thus, checking either it is at TDC or BDC. If it is at BDC, it will take it to TDC. The dispense solenoid (Q0.3) also comes on and hence, dispensing the washer onto the conveyor. This will also allow conveyor belt to move (Q0.4). When it reaches motorised gauge, the conveyor stops to	Refer to network 1 network 4 (Q0.2)HEIGHT_MOTOR, network 6 (Q0.3)DISPENSER and network 7 (Q0.4)CONVEYOR

		performance the height measurement of the washer.	
5	Switch ON I0.3 OPTO_1. Simulating the detection of the incoming washer.	This network will simulate the detection of the incoming washer and hence, alerting the motor to bring height motorised arm to it BDC (by calling subroutine 1, SBR_1) for the height measurement and return it to the TDC (SBR_2). To successfully perform height measurement, T100, T36 timers or RUN_ON_LATCH are needed to correctly position washer at the right position, at the right time and allow nought time to perform height measurement operation	Refer to the network 8, Network 11 and network 12
6	Switch ON I0.4 OPTO_2. Simulating the detection of the material type. That is, either metal or plastic	This network will determine the type of material (metal/plastic). This is accomplished by latching its flag (AT_OPTO_2). It was also assumed previously that the height was gauge as ok.	Refer to network 13
7	Press I1.0 (METAL_PLASTIC). Simulation for directing the metal washer into metal bin.	This network will direct the metal washer directly to metal bin (Q0.6). By pressing I1.0, it will set the metal flag to be HIGH. Thus,	Refer to network 20 (METAL_FLAG) Network 24 (FLIPPER_1_OFF) And network 25 (FLIPPER_1)

		energising the metal bin within a period of time determined by T33 timer.	
8	Press stop I0.6 (STOP_PRESS) Stopping the conveyor system	By press stop button, the system will be halted thus, clearing all the flags used (reset).	Refer to network 2

Second Scenario (case 2)

Assume the washer is metal and the height of the washer is either too small or too big.

Sequence Number	Operator Action	Expected Outcomes	Comments
1	Switch on I0.0 (TDC_OPTO). Simulation of the first subroutine	This subroutine will be responsible of initial checking either the height motorised gauge is at the TDC or BDC. Pressing I0.0 will latch and set M20.0 to be high thus setting flag TDC_MOTOR_RUN (M20.1). TDC_MOTOR_RUN Flag is set to run the motor only at the start and should not run again until the main program has being halted. TDC_FLAG (M20.2) target is also set to take the height measurement gauge to it uppermost provide that the hight motorised	Refer SBR_0 first subroutine.

		gauge was found to be at the BDC. Finally STILL_IN_SUB0 (M20.3) flag is set to stop other subroutines from running until it is finish..	
2	Switch on I0.1 (BDC_OPTO) simulation of the second subroutine	The expected outcome for this subroutine is to lower the height motorised gauge to it BDC. Normally On special memory (SM0.0) is used to Reset C1 (TDC_COUNTER) so that it can be used for the next incoming washer. Pressing I0.1 will latch M12.0 to be ON (HIGH) until it target (flag) is clear. This is achieved by setting the following flags I.E BDC_MOTOR_RUN flag which lower the height motorised gauge to it lowest part and STILL_IN_SUB1 flag which stop other subroutines from running until this subroutine is finished. Most importantly, Height Ok flag (HGHT_OK) is also set to indicate if the height of the washer is within the predefined range of limits. If the HGHT_OK is high, then	Refer to SBR_1 second subroutine

		the detection of the material type will be determined. This will be discussed further in sequence number 6. if the HGHT_OK is not ok, then there is not need to do any more detection and hence, the washer will be directed to the dumping bin,	
3	Switch on I0.0 (TDC_OPTO). Simulation of the third subroutine	This subroutine is responsible of returning the height motorised gauge to it uppermost (TDC). SM0.0 special memory is used to reset C3 so that it can be recall for the next cycle of incoming washer. To successfully return the Height motorised gauge to it TDC, TDC_MOTOR_RUN flag is set as well setting another flag (STILL_IN_SUB_2) to avoid other subroutines from interrupting while this subroutine is running until finished.	Refer to third subroutine (SBR_2)
4	Press I0.5 (START_PRESS), a normal starting of the conveyor	This will energise height motorised gauge (Q0.2) thus, checking either it is at TDC or BDC. If it is at BDC, it will take it to TDC. The dispense solenoid (Q0.3) also comes on and hence, dispensing	Refer to network 1, network 4 (Q0.2)HEIGHT_MOTOR, network 6 (Q0.3)DISPENSER and network 7 (Q0.4)CONVEYOR

		the washer onto the conveyor. This will also allow conveyor belt to move (Q0.4). When it reaches motorised gauge, the conveyor stops to performance the height measurement of the washer.	
5	Switch ON I0.3 OPTO_1. Simulating the detection of the incoming washer.	This network will simulate the detection of the incoming washer and hence, alerting the motor to bring height motorised arm to it BDC (by calling subroutine 1, SBR_1) for the height measurement and return it to the TDC (SBR_2). To successfully perform height measurement, T100, T36 timers or RUN_ON_LATCH are needed to correctly position washer at the right position, at the right time and allow nought time to perform height measurement operation	Refer to the network 8, Network 11 and network 12
6	Switch ON I0.4 OPTO_2. Simulating the detection of the material type. That is, either metal or plastic	This network will determine the type of material (metal/plastic). This is accomplished by latching its flag (AT_OPTO_2). It was also assumed previously that the height was gauge as ok.	Refer to network 13

7	Press stop I0.6 (STOP_PRESS) Stopping the conveyor system	By press stop button, the system will be halted thus clearing all the flags used(reset) and conveyor will be on OFF motion	Refer to network 2
---	---	--	--------------------

Third Scenario (case 3)

Assume the washer is plastic and the height is within the defined height limits

Sequence Number	Operator Action	Expected Outcomes	Comments
1	Switch on I0.0 (TDC_OPTO). Simulation of the first subroutine	This subroutine will be responsible of initial checking either the height motorised gauge is at the TDC or BDC. Pressing I0.0 will latch and set M20.0 to be high thus setting flag TDC_MOTOR_RUN (M20.1). TDC_MOTOR_RUN Flag is set to run the motor only at the start and should not run again until the main program has being halted. TDC_FLAG (M20.2) target is also	Refer SBR_0 first subroutine.

		<p>set to take the height measurement gauge to it uppermost provide that the hight motorised gauge was found to be at the BDC. Finally STILL_IN_SUB0 (M20.3) flag is set to stop other subroutines from running until it is finish.</p>	
2	<p>Switch on I0.1 (BDC_OPTO) simulation of the second subroutine</p>	<p>The expected outcome for this subroutine is to lower the height motorised gauge to it BDC. Normally On special memory (SM0.0) is used to Reset C1 (TDC_COUNTER) so that it can be used for the next incoming washer. Pressing I0.1 will latch M12.0 to be ON (HIGH) until it target (flag) is clear. This is achieved by setting the following flags I.E BDC_MOTOR_RUN flag which lower the height motorised gauge to it lowest part and STILL_IN_SUB1 flag which stop other subroutines from running until this subroutine is finished. Most importantly, Height Ok flag (HGHT_OK) is also set to indicate if the</p>	<p>Refer to SBR_1 second subroutine</p>

		height of the washer is within the predefined range of limits. If the HGHT_OK is high, then the detection of the material type will be determined. This will be discussed further in sequence number 6. if the HGHT_OK is not ok, then there is not need to do any more detection and hence, the washer will be directed to the dumping bin,	
3	Switch on I0.0 (TDC_OPTO). Simulation of the third subroutine	This subroutine is responsible of returning the height motorised gauge to it uppermost (TDC). SM0.0 special memory is used to reset C3 so that it can be recall for the next cycle of incoming washer. To successfully return the Height motorised gauge to it TDC, TDC_MOTOR_RUN flag is set as well setting another flag (STILL_IN_SUB_2) to avoid other subroutines from interrupting while this subroutine is running until finished.	Refer to third subroutine (SBR_2)
4	Press I0.5 (START_PRESS), a normal starting of the conveyor	This will energise height motorised gauge (Q0.2) thus, checking either it is at TDC or BDC. If it is at BDC, it	Refer to network 1 network 4 (Q0.2)HEIGHT_MOTOR, network 6 (Q0.3)DISPENSER and

		will take it to TDC. The dispense solenoid (Q0.3) also comes on and hence, dispensing the washer onto the conveyor. This will also allow conveyor belt to move (Q0.4). When it reaches motorised gauge, the conveyor stops to performance the height measurement of the washer.	network 7 (Q0.4)CONVEYOR
5	Switch ON I0.3 OPTO_1. Simulating the detection of the incoming washer.	This network will simulate the detection of the incoming washer and hence, alerting the motor to bring height motorised arm to it BDC (by calling subroutine 1, SBR_1) for the height measurement and return it to the TDC (SBR_2). To successfully perform height measurement, T100, T36 timers or RUN_ON_LATCH are needed to correctly position washer at the right position, at the right time and allow nought time to perform height measurement operation	Refer to the network 8, Network 11 and network 12
6	Switch ON I0.4 OPTO_2. Simulating the detection of the material type. That is, either metal or plastic	This network will determine the type of material (metal/plastic). This is accomplished by latching its flag	Refer to network 13

		(AT_OPTO_2). It was also assumed previously that the height was gauge as ok.	
7	Press I1.0 (METAL_PLASTIC). Simulation for directing the plastic washer into plastic bin.	This network will direct the plastic washer directly to plastic bin (Q0.5). By pressing I1.0, it will set the plastic flag to be HIGH. Thus, energising the plastic bin within a period of time determined by T97 timer.	Refer to network 20 (METAL_FLAG) Network 21 (FLIPPER_2_OFF) And network 22 (FLIPPER_2)
8	Press stop I0.6 (STOP_PRESS) Stopping the conveyor system	By press stop button, the system will be halted thus, clearing all the flags used (reset).	Refer to network 2

Fourth Scenario (case 4)

Assume the washer is plastic and the height of the washer is either too small or too big.

Sequence Number	Operator Action	Expected Outcomes	Comments
1	Switch on I0.0 (TDC_OPTO). Simulation of the first subroutine	This subroutine will be responsible of initial checking either the height motorised gauge is at the TDC or BDC. Pressing I0.0 will latch and set M20.0 to be high thus setting flag TDC_MOTOR_RUN (M20.1). TDC_MOTOR_RUN	Refer SBR_0 first subroutine.

		<p>Flag is set to run the motor only at the start and should not run again until the main program has being halted. TDC_FLAG (M20.2) target is also set to take the height measurement gauge to it uppermost provide that the hight motorised gauge was found to be at the BDC. Finally STILL_IN_SUB0 (M20.3) flag is set to stop other subroutines from running until it is finish.</p>	
2	<p>Switch on I0.1 (BDC_OPTO) simulation of the second subroutine</p>	<p>The expected outcome for this subroutine is to lower the height motorised gauge to it BDC. Normally On special memory (SM0.0) is used to Reset C1 (TDC_COUNTER) so that it can be used for the next incoming washer. Pressing I0.1 will latch M12.0 to be ON (HIGH) until it target (flag) is clear. This is achieved by setting the following flags I.E BDC_MOTOR_RUN flag which lower the height motorised gauge to it lowest part and STILL_IN_SUB1 flag</p>	<p>Refer to SBR_1 second subroutine</p>

		<p>which stop other subroutines from running until this subroutine is finished. Most importantly, Height Ok flag (HGHT_OK) is also set to indicate if the height of the washer is within the predefined range of limits. If the HGHT_OK is high, then the detection of the material type will be determined. This will be discussed further in sequence number 6. if the HGHT_OK is not ok, then there is not need to do any more detection and hence, the washer will be directed to the dumping bin,</p>	
3	<p>Switch on I0.0 (TDC_OPTO). Simulation of the third subroutine</p>	<p>This subroutine is responsible of returning the height motorised gauge to it uppermost (TDC). SM0.0 special memory is used to reset C3 so that it can be recall for the next cycle of incoming washer. To successfully return the Height motorised gauge to it TDC, TDC_MOTOR_RUN flag is set as well setting another flag (STILL_IN_SUB_2) to avoid other subroutines from interrupting while</p>	<p>Refer to third subroutine (SBR_2)</p>

		this subroutine is running until finished.	
4	Press I0.5 (START_PRESS), a normal starting of the conveyor	This will energise height motorised gauge (Q0.2) thus, checking either it is at TDC or BDC. If it is at BDC, it will take it to TDC. The dispense solenoid (Q0.3) also comes on and hence, dispensing the washer onto the conveyor. This will also allow conveyor belt to move (Q0.4). When it reaches motorised gauge, the conveyor stops to performance the height measurement of the washer.	Refer to network 1, network 4 (Q0.2)HEIGHT_MOTOR, network 6 (Q0.3)DISPENSER and network 7 (Q0.4)CONVEYOR
5	Switch ON I0.3 OPTO_1. Simulating the detection of the incoming washer.	This network will simulate the detection of the incoming washer and hence, alerting the motor to bring height motorised arm to it BDC (by calling subroutine 1, SBR_1) for the height measurement and return it to the TDC (SBR_2). To successfully perform height measurement, T100, T36 timers or RUN_ON_LATCH are needed to correctly position washer at the right position, at the right time and allow nought time to perform	Refer to the network 7, network8, Network 11 and network 12

		height measurement operation	
6	Switch ON I0.4 OPTO_2. Simulating the detection of the material type. That is, either metal or plastic	This network will determine the type of material (metal/plastic). This is accomplished by latching its flag (AT_OPTO_2). If the AT_OPTO_2 is high and the height measurement was ok (height within the specified range) then stop the conveyor to perform metal/plastic detection for about a period of two second which is timed out by T96. This sets HGHT_Ok flag if these conditions are satisfied and stop the conveyor for the period of 2 seconds. It was also assumed previously that the height was gauge as ok.	Refer to network 13
7	Press stop I0.6 (STOP_PRESS) Stopping the conveyor system	By press stop button, the system will be halted thus clearing all the flags used(reset) and conveyor will be on OFF motion	Refer to network 2

